第一篇　图　论

第一章　基本概念

本章开始介绍图论的若干基本概念，也将提到包含这些概念的某些结果。这些结果既可用来说明概念，也可用来给读者指出一些证明图论定理常用的技巧。

1.1　基本定义

一个图$G=(V,E)$由两个集组成：有限集V中的元素称为顶点，有限集E中的元素称为边。每条边连在一对顶点之间。若图G的边是连在有序顶点对之间[+]，则G称为有向图或定向图。反之，G称为无向图或非定向图。本书的前四章讨论无向图。

通常用符号v_1，v_2，v_3，…表示图的顶点，用符号e_1，e_2，e_3，…表示图的边。与边e_i相联接的顶点v_i和v_j叫做e_i的端点。因而，e_i可表示为$e_i=(v_i, v_j)$。应该指出，尽管集E中的元素是各不相同的，但同一对端点间却可有一条以上在E中的边。这些具有同一对端点的边称为平行边。有时，一条边的两个端点是同一的，即$e_i=(v_i, v_i)$，则边e_i称为在顶点v_i处的自环。不含平行边和自环的图称为简单图。如果图的顶点集中含有n个元素，则图的阶数为n。

无边的图称为零图。无顶点（也就无边）的图称为空图。

图常用一个曲线图形来表示，图的顶点常用小圆点或圆圈来表示，边常用连接两个代表端点的小圆点或圆圈之间的一条线段来表示。例如，若

$$V=\{v_1, v_2, v_3, v_4, v_5, v_6\}$$

和

$$E=\{e_1, e_2, e_3, e_4, e_5\}$$

而

$$e_1=(v_1, v_2)$$
$$e_2=(v_3, v_4)$$
$$e_3=(v_5, v_6)$$
$$e_4=(v_1, v_2)$$
$$e_5=(v_5, v_6)$$

则图$G=(V, E)$如图1.1所示。其中，e_1和e_4是平行边，e_5是自环。

边对应其端点叫做边关联于顶点。若两个顶点是某个边的端点，这两个顶点就叫做邻

+指连在一条边两端的一对顶点(v_i, v_j)是有次序的，以v_i为始点，v_j为终点，即边具有一定的方向。边(v_i, v_j)和边(v_j, v_i)是E中两个不同的元素。——译者注

接。两个边有同一端点，这两个边也叫做邻接。

例如在图1.1中，边e_1与顶点v_1和v_2关联，v_1和v_4是两个邻接顶点。而e_1和e_2是两条邻接边。

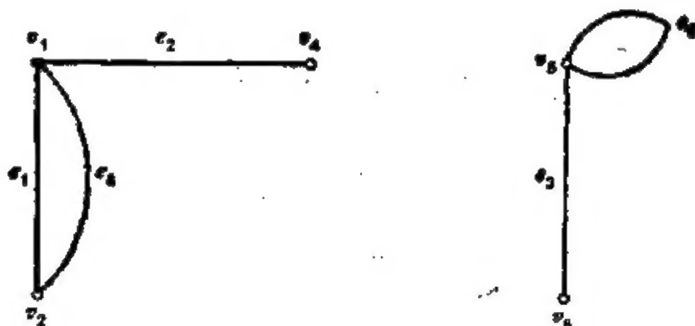


图 1.1 图$G=(V,\ E)$。$V=\{v_1,\ v_2,\ v_3,\ v_4,\ v_5,\ v_6\}$；$E=\{e_1,\ e_2,\ e_3,\ e_4,\ e_6\}$。

与顶点v_i关联的边数叫做v_i的度，并记为$d(v_i)$，顶点的度数也叫做顶点的价。度数为1的顶点叫做悬挂顶点，与悬挂顶点关联的那条边叫做悬挂边。度数为0的顶点叫做孤立顶点。按照定义，顶点v_i上的每个自环都使v_i的度数增加2。$\delta(G)$和$\Delta(G)$分别表示图G中顶点度数的最小值和最大值。

在图1.1所示图G中，

$$d(v_1)=3$$
$$d(v_2)=2$$
$$d(v_3)=0$$
$$d(v_4)=1$$
$$d(v_5)=3$$
$$d(v_6)=1$$

图中，v_3是孤立顶点，v_4和v_6是悬挂顶点，e_2是悬挂边。从图G中可证实，所有各顶点的度数之和为10，而边数为5。即图G的顶点的度数之和为边数的两倍，是个偶数。还可进一步证实，度数为奇数的顶点也是偶数个。这个有趣的结果并不限于图1.1中的图，事实上，它对任何图都适用，正如下面定理所述。

定理1.1[+] 图G的所有顶点度数之和为$2m$，m是G的边数。

证明 因为每条边都与两个顶点关联，并使图G的度数之和增添2，所以由m条边组成的图的顶点度数之和为$2m$。 □

定理1.2[++] 在任一图中，度数为奇数的顶点有偶数个。

证明 设图G共有n个顶点。不失一般性，令前r个顶点v_1，v_2，…，v_r的度数都是偶数，其余$(n-r)$个顶点的度数都为奇数。则

$$\sum_{i=1}^{n} d(v_i)=\sum_{i=1}^{r} d(v_i)+\sum_{i=r+1}^{n} d(v_i) \tag{1.1}$$

[+]定理1.1被称为图论中的第一定理，是图论创始人欧拉在1736年提出的。——译者注

[++]定理1.2被称为握手引理，即宴会上宾客见面时要互相握手，握手奇次数的宾客必为偶数。——译者注

根据定理1.1，图G所有顶点度数的总和是偶数，即式（1.1）左边是个偶数。右边第一求和式是r个偶数的总和，当然是偶数，因此右边第二求和式也必然是偶数。因为第二求和式中每项都是奇数，所以第二求和式中项数必为偶数，即度数为奇数的顶点数（$n-r$）必为偶数。 □

1.2 子图和补图

考察图$G=(V, E)$。如果V'和E'分别是V和E的子集，且边(v_i, v_j)属于E'，仅当v_i，v_j属于V'，那么$G'=(V', E')$就是G的子图。如果E'是E的真子集或V'是V的真子集，那么G'就叫做G的真子图。如果图G的所有顶点都出现在G的子图G'中，那么G'就叫做G的生成子图。

例如，请看图1.2(*a*)所示图G。图1.2(*b*)所示图G'是G的子图，其顶点集是$\{v_1, v_2, v_4, v_5\}$。事实上，它是G的真子图。图1.2(*c*)是G的生成子图。

子图中某些顶点可以是孤立顶点。例如，图1.2(*d*)所示图G''就是G的具有一个孤立顶点的子图。

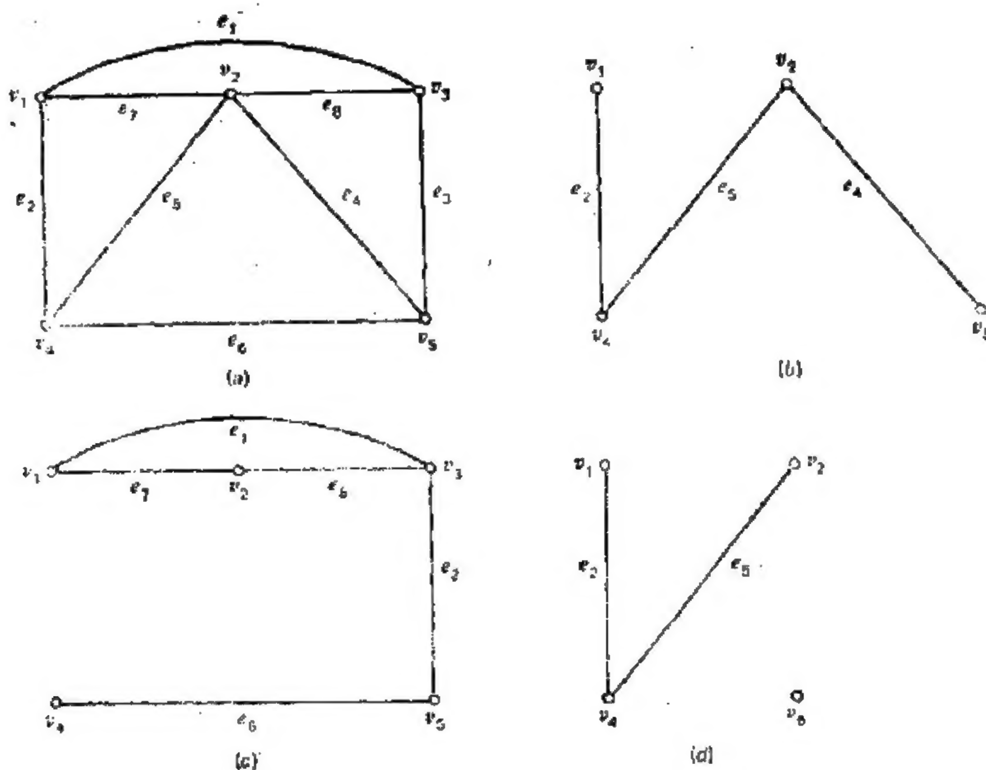


图 1.2 图和某些子图

(*a*)图G；(*b*)子图G'；(*c*)子图G''；(*d*)子图G''。

如果图G的子图$G'=(V', E')$不含有孤立顶点，那么从子图的定义可以看出，V'中每个顶点都是E'中某条边的端点。于是，在这种情形下，E'唯一地确定了V'及子图G'。子图G'就叫做G在边集E'上的导出子图（或简称为G的边-导出子图），记为$\langle E'\rangle$。

注意，$\langle E'\rangle$的顶点集V'是V中含E'所有边端点的最小子集。图1.2(*b*)和1.2(*c*)所示

子图 G' 和 G'' 是图1.2(a)图 G 的边-导出子图，而图1.2(d)所示图 G''' 就不是边-导出子图。

下面定义顶点-导出子图。

令V'是图$G=(V, E)$的顶点集V的子集。如果E'是E的子集，且边(v_i, v_j)属于E'，当且仅当v_i和v_j属于V'，那么子图$G'=(V', E')$就叫做G在顶点集V'上的导出子图（或简称为G的顶点-导出子图）。也就是说，如果v_i和v_j属于V'，那么E中每一条以v_i和v_j为端点的边都属于E'。注意，在这种情形下，V'完全确定了E'及子图G'。于是，顶点-导出子图$G'=(V', E')$就简记为$\langle V'\rangle$。作为例子，图1.3是图1.2(a)图G的顶点-导出子图。

图 1.3 图1.2(a)图G的顶点-导出子图

注意，在顶点集V'上的顶点-导出子图$\langle V'\rangle$的边集E'是E的最大子集，从而E'的所有边的端点都在V'中。

图G的子图G'叫做G对于某性质P的最大子图，如果G'具有性质P，且不是G的其它任何具有性质P的子图的真子图。

图G的子图G'叫做G对于某性质P的最小子图，如果G'具有性质P，且G的其它具有性质P的子图不是G'的真子图。

对于某性质的集合的最大和最小子集同样定义。

例如，图$G=(V, E)$的边-导出子图$\langle E'\rangle$的顶点集V'是V中含E'所有边端点的最小子集。另一方面，顶点-导出子图$\langle V'\rangle$的边集E'是E的所有端点属于V'的边的最大子集。

接下来，我们看到图G的"片"（1.4节）是G的最大"连通"子图，连通图G的生成树（第二章）是G的最小"连通"生子成图。

下面定义图的补图。

图$\bar{G}=(V, E')$叫做简单图$G=(V, E)$的补图，当且仅当E'中的边(v_i, v_j)不在E中。即两个顶点v_i和v_j在$\bar{G}$中是相邻的，而它们在G中是不相邻的。图1.4给出了一个图和它的补图。另一个例子，考察1.5(a)所示图G。图中每对顶点间都有一条边，所以G的补图$\bar{G}$中的任意顶点对间均没有边，即$\bar{G}$仅包含孤立顶点，这已在图1.5(b)中示出。

令$G'=(V', E')$是图$G=(V, E)$的子图，G的子图$G''=(V, E-E')$叫做G'在G中的补图。例如，在图1.2中，子图G''是G'在图G中的补图。

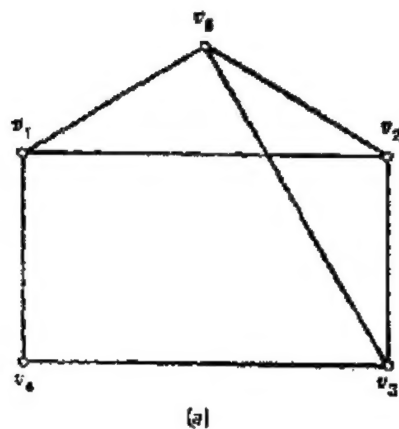
下面的例子解释前面介绍的几个概念。

假定要证明下面的问题：

在任意有六个人参加的宴会上，有三个人互相认识，或者有三个人互不相识。

用图的顶点代表人，用连接相应顶点的边表示人们之间相识关系，容易看出，上面的论断可陈述如下：

在任意六个顶点的简单图中，有三个相邻顶点，或有三个不相邻顶点。



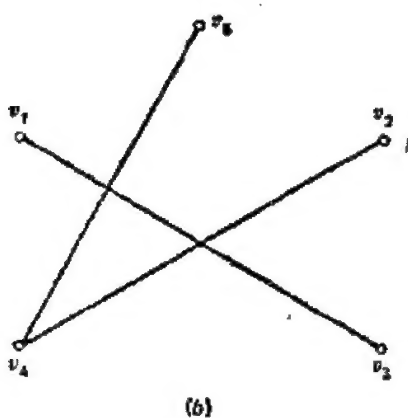
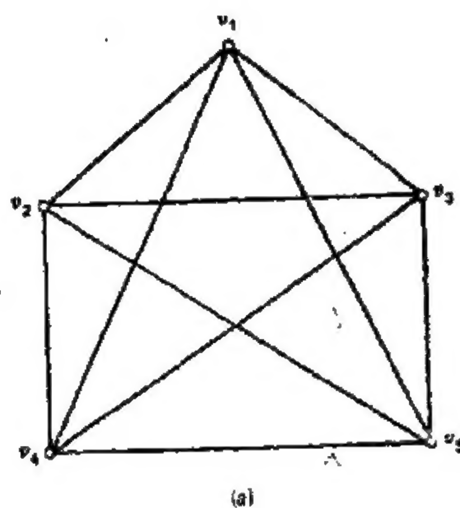


图 1.4 图及其补图
(a)图G；(b)图$\overline{G}$，G的补图。



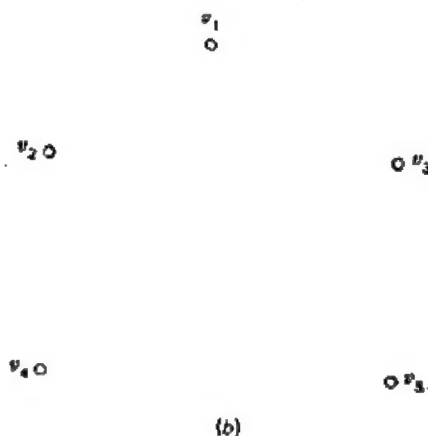


图 1.5 图及其补图。
(a)图G；(b)图G，$\overline{G}$的补图。

从补图的定义出发，可以看出上述说法等价于下面的叙述：

对任意六个顶点的简单图G，G或$\overline{G}$包含了三个相邻顶点。

为了证明这个结论，可如下进行：

考察六个顶点简单图G的任意顶点v。注意，如果v在G中不与三个顶点相邻，那么它一定在$\overline{G}$中与三个顶点相邻。不失一般性，可假定v与G中某三个顶点v_1、v_2和v_3相邻。如果这三个顶点中任意两个在G中相邻，比如是v_1和v_2，那么v、v_2和v_1在G中就是相邻的，于是论断得证。

如果v_1、v_2和v_3三个顶点在G中任意两个都是不相邻的，那么就意味着v_1、v_2和v_3在G中是互不相邻的。于是，由补图的定义可知，v_1、v_2和v_3在$\overline{G}$中是相邻的，论断再次得证。

1.3 通道、轨迹、路径和回路

图G=(V, E)中，通道是顶点和边的有限交错序列v_0，e_1，v_1，e_2，…，v_{k-1}，e_k，

v_k，它以顶点起始和结束，且v_{i-1}和v_i是边e_i的端点，$1 \leqslant i \leqslant k$。反之，通道可以看作顶点的有限序列$v_0$，$v_1$，$v_2$，…，$v_k$，其中$(v_{i-1}, v_i)$是$G$中的边，$1 \leqslant i \leqslant k$。这样的通道通常叫做$v_0$-$v_k$通道，$v_0$、$v_k$叫做这条通道的终点或端点，其它所有顶点叫做这条通道的内部顶点。注意，通道中的边和顶点可以出现不止一次。

通道的两个端点不同时，通道是开的，否则是闭的。

在图1.6的图G中，序列v_1，e_1，v_2，e_2，v_3，e_8，v_6，e_9，v_5，e_7，v_3，e_{11}，v_6是开通道，而序列v_1，e_1，v_2，e_3，v_5，e_7，v_3，e_2，v_2，e_1，v_1是闭通道。

通道所有的边都不相同时，叫做轨迹。轨迹的端点是相异的，叫做开轨迹；否则，叫做闭轨迹。在图1.6中，v_1，e_1，v_2，e_2，v_3，e_8，v_6，e_{11}，v_3是开轨迹，而v_1，e_1，v_2，e_2，v_3，e_7，v_5，e_3，v_2，e_4，v_4，e_5，v_1是闭轨迹。

当开轨迹的所有顶点都不相同时，叫做路径（路）。

而闭轨迹除端点外所有顶点都不相同时，叫做回路。

例如，在图1.6中，序列v_1，e_1，v_2，e_2，v_3是路径，而序列v_1，e_1，v_2，e_3，v_5，e_6，v_4，e_5，v_1是回路。

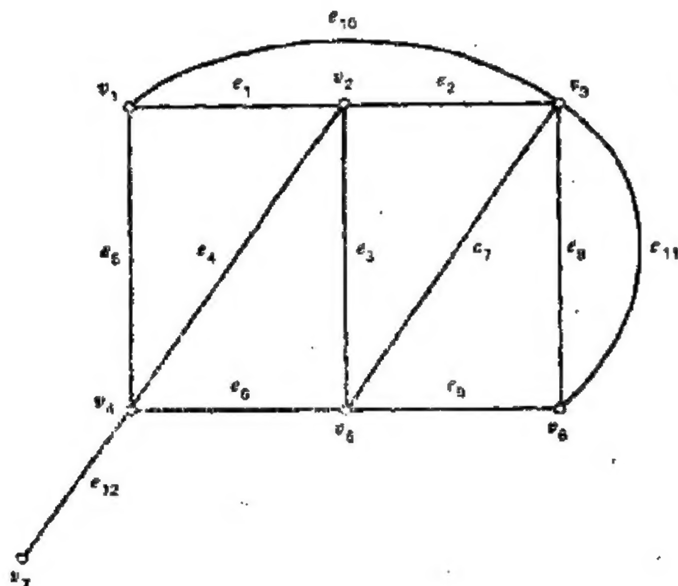


图 1.6　图G

如果图G中有含某条边的回路，则这条边就叫做回路边。否则，这条边就叫做非回路边。图1.6中，除了e_{12}，所有的边都是回路边。

路径中边的数目叫做路径长度。回路的长度也同样定义。

应当注意路径和回路的下列性质：

1.路径中，每个不是端点的顶点度数等于2；端点度数等于1。

2.回路中，每个顶点的度数为2，即偶度数，这个命题的逆命题，即每个顶点都具有偶度数的子图的边构成回路，是不正确的。更一般的问题在第三章讨论。

3.路径中，顶点数等于边数加一；而回路中，边数等于顶点数。

1.4 图的连通性和片

连通性是图论中的一个重要概念。

如果图G中存在$v_i—v_j$路径，则两个顶点v_i和v_j在G中就叫做连通的。一个顶点与它本身是连通的。

如果G中每对顶点之间都存在一条路径，则图G是连通的。

例如，图1.6的图就是连通的。

考察不连通图$G=(V,E)$。G的顶点集V可以划分*为子集V_1，V_2，…，V_p，以使得顶点-导出子图$\langle V_i\rangle$是连通的，$i=1, 2, \cdots, p$，且子集V_i中的顶点不与V_j中任何顶点连通，$j\neq i$。子图$\langle V_i\rangle$叫做G的片，$i=1, 2, \cdots, p$。可以看出，图G的片是G的最大连通子图：即G的片不是G的其它任何连通子图的真子图。

例如，图1.7的图G是不连通的，它的四个片的顶点集分别为$\{v_1,v_2,v_3\}$，$\{v_4,v_5\}$，$\{v_6,v_7,v_8\}$以及$\{v_9\}$。

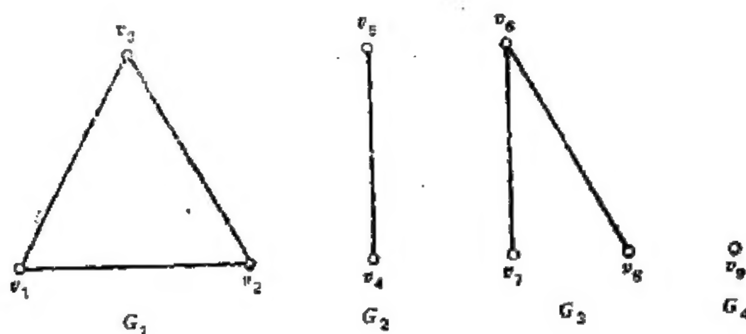


图 1.7 具有片G_1，G_2，G_3和G_4的图G

注意，由定义，一个顶点与它本身连通，所以孤立顶点本身也应该当作片来处理。而且，如果图G是连通的，那么它仅有一个与其本身相同的片。

下面研究连通图的几个性质。

定理 1.3 在连通图中，任何两条最长路径有公共顶点。

证明 考察连通图G中任何两条最长路径P_1和P_2。用顶点序列v_0，v_1，v_2，…，v_k表示P_1，用顶点序列v_0'，v_1'，v_2'，…，v_k'表示P_2。

假定P_1和P_2没有公共顶点。因为图是连通的，所以对于某个$i(0\leqslant i\leqslant k)$和某个$j(0\leqslant j\leqslant k)$，则存在$v_i—v_j'$路径$P_0$，并使得$P_0$除$v_i$和$v_j'$外所有顶点都与$P_1$和$P_2$的顶点不同。路径$P_1$、$P_2$和$P_0$示于图1.8中。令

$$t_1=v_0—v_i\text{路径}P_{11}\text{的长度}$$
$$t_2=v_i—v_k\text{路径}P_{12}\text{的长度}$$
$$t_1'=v_0'—v_j'\text{路径}P_{21}\text{的长度}$$

---

* 集合V可划分为子集V_1，V_2，…，V_p，如果$V_1\cup V_2\cup\cdots\cup V_p=V$，对所有$i$和$j$，$i\neq j$，有$V_i\cap V_j=\phi$。那么$\{V_1, V_2, \cdots, V_p\}$叫做$V$的划分。

$$t_2' = v_j' - v_k' \text{路径} P_{22} \text{的长度}$$

$$t_0 = \text{路径} P_0 \text{的长度}$$

路径P_{11}，P_{12}，P_{21}和P_{22}也在图1.8中示出。注意，

$$t_1 + t_2 = t_1' + t_2' = G\text{中最长路径的长度}$$

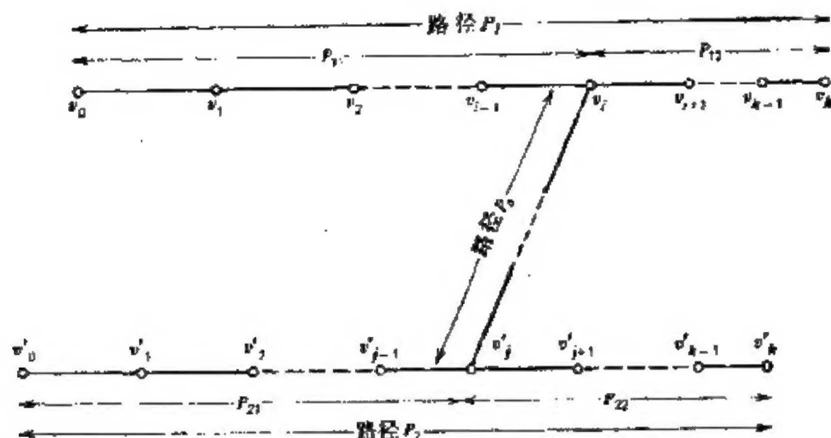


图 1.8 路径P_1，P_2和P_0

以及

$$t_0 > 0$$

不失一般性，令

$$t_1 \geqslant t_2$$

以及

$$t_1' \geqslant t_2'$$

以使得

$$t_1 + t_1' \geqslant t_1 + t_2 = t_1' + t_2'$$

现在可以证明：路径P_{11}、P_0和P_{21}在一起构成了$v_0 - v_0'$路径，其长度$t_1 + t_1' + t_0 > t_1 + t_2$，因为$t_0 > 0$。这与$t_1 + t_2$是G中最长路径的长度相矛盾。 □

下面的定理是非常有用的，在下章的讨论中经常用到。在这个定理以及本书的其它定理中，当我们很明显指的是集而不是元素时，就把$\{x\}$简记为x。

定理1.4 如果图$G = (V, E)$是连通的，那么在其中移去一条回路边e所得到的图$G' = (V, E - e)$仍是连通的。 □

此定理的证明留作练习。

1.5 图的运算

本节介绍几种图的运算方法。前三种是属于两个图的二元运算，后四种是一元运算，即对单图的运算。

设有两个图$G_1 = (V_1, E_1)$和$G_2 = (V_2, E_2)$。图G_1和G_2的并（用$G_1 \cup G_2$表示）是图$G_3 = (V_1 \cup V_2, E_1 \cup E_2)$，即$G_3$的顶点集是$V_1$和$V_2$的并，而$G_3$的边集是$E_1$和$E_2$

的并。

例如，在图1.9中，G_1、G_2以及它们的并G_3分别由图1.9(a)、(b)、(c)表示。

图G_1和G_2的交（用$G_1 \cap G_2$表示）是图$G_4=(V_1 \cap V_2, E_1 \cap E_2)$，即$G_4$的顶点集仅包含$G_1$和$G_2$的公共顶点，而$G_4$的边集仅包含$G_1$和$G_2$的公共边。

图1.9(a)、(b)的图G_1和G_2的交，如图1.9(d)所示。

图G_1和G_2的环和（用$G_1 \oplus G_2$表示）是边集$E_1 \oplus E_2$的导出图G_5。换言之，G_5无孤立顶点，它仅含有在G_1中或在G_2中而不是两者都有的边。图1.9(a)和(b)的环和，如图1.9(e)所示。

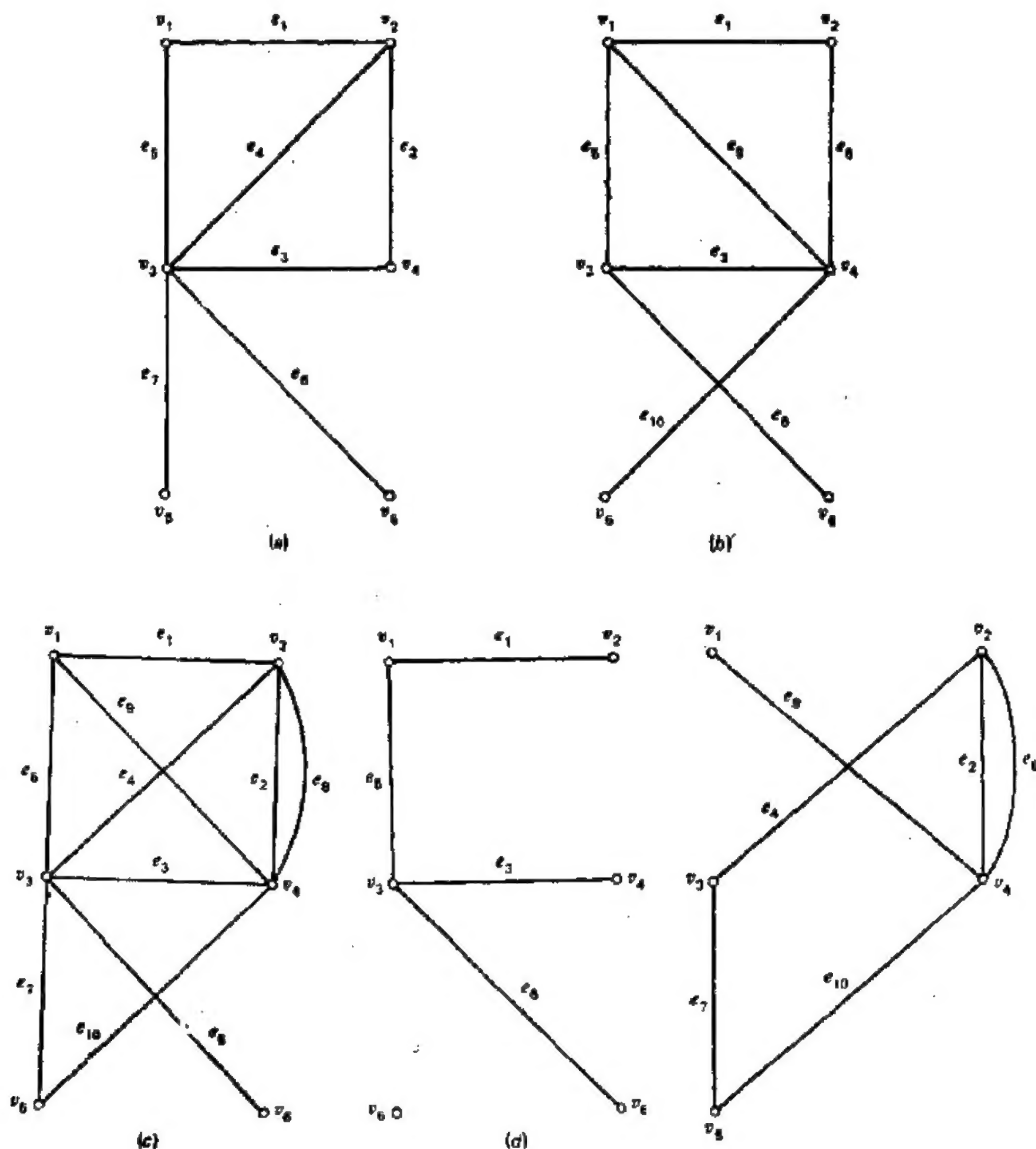


图 1.9 图的运算：并、交、环和运算
(a)图G_1；(b)图G_2；(c)$G_1 \cup G_2$；(d)$G_1 \cap G_2$；(e)$G_1 \oplus G_2$。

显而易见，上述三种运算是符合交换律的，即

$$G_1\cup G_2=G_2\cup G_1$$
$$G_1\cap G_2=G_2\cap G_1$$
$$G_1\oplus G_2=G_2\oplus G_1$$

还须指出，这些运算是二元的，即相对于两个图的运算而言。当然，这些运算的定义显然可扩展到两个图以上。

下面将讨论图的四种一元运算。

顶点去除 设v_i是图$G=(V,E)$的一个顶点，则$G-v_i$是以$V-v_i$为顶点集的G的导出子图；即$G-v_i$是从G中去除顶点v_i以及和v_i关联的所有边后形成的图。

边去除 设e_i是图$G=(V,E)$的一条边，则$G-e_i$是从G中去除边e_i后所得的子图。应注意，边e_i的端点并不从G中去除。

从图上去除一组顶点或一组边，就等于把顶点或边一个接一个地移去。

若图$G_1=(V',E')$是图$G=(V,E)$的一个子图，则$G-G_1$指的是图$G'=(V',E-E')$。$G-G_1$叫做G_1在G中的补图。

顶点去除与边去除示于图1.10。

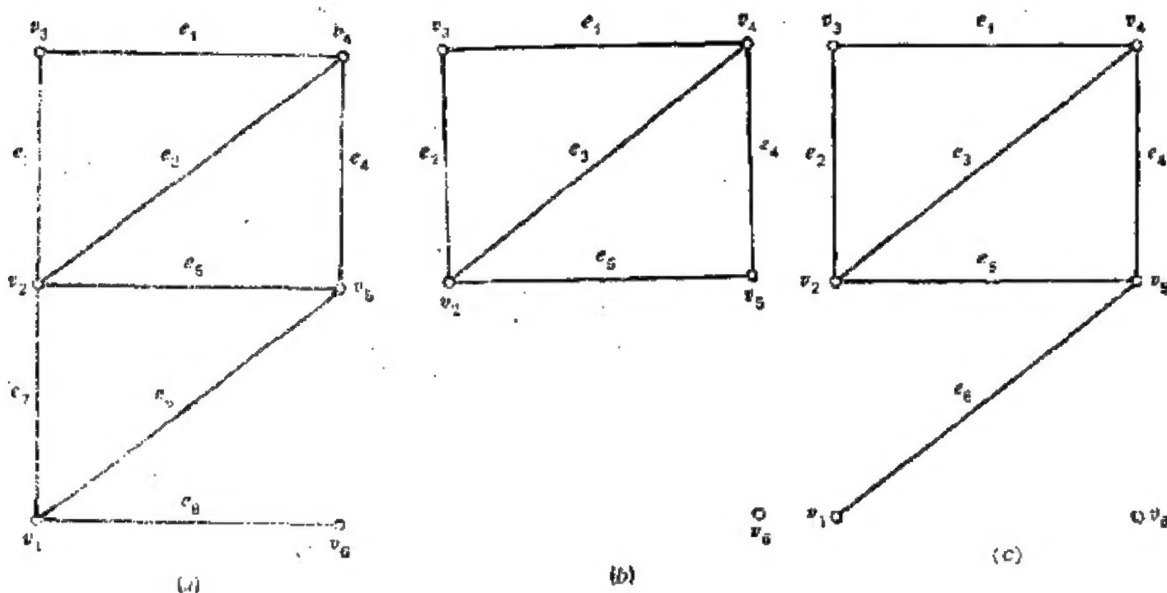


图 1.10 图的顶点去除和边去除运算
(a)图G；(b)$G-v_1$；(c)$G-\{e_7, e_8\}$。

下面介绍图的短接或合一运算，这对于电气工程师是比较熟悉的。

短接（合一） 图G的一对顶点v_i和v_j短接（合一），指的是这两个顶点由一新顶点代替，原先与v_i和v_j关联的所有边现在都与新顶点关联。例如，将图1.11(*a*)中顶点v_3和v_4短接后，就变成图1.11(*b*)。

收缩 边的收缩系指将边去除，并把它的端点短接。当图G经过一系列收缩能形成H时，则称G可收缩成H。例如，从图1.11(*a*)图G中收缩边e_1和e_5，就可得到图1.11(*c*)。

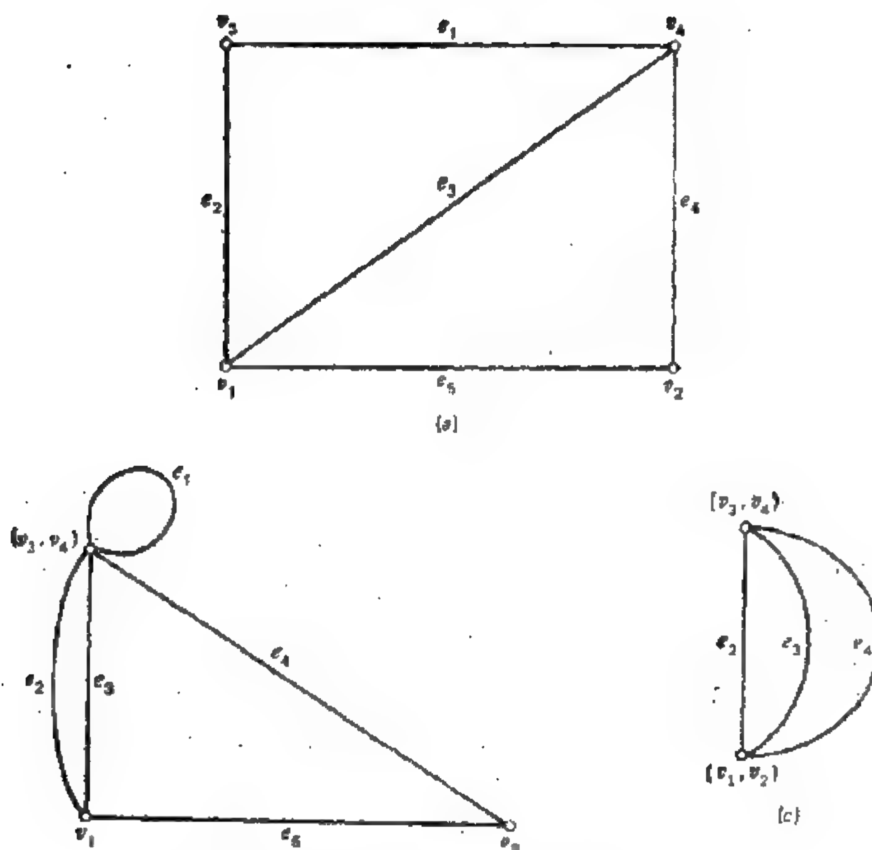


图 1.11　图的短接和收缩

(a)图G_1　(b)v_3和v_4短接后的图G_1　(c)e_1和e_5收缩后的图G_1

1.6 特殊图

下面介绍图论中经常出现的几种特殊图。

完全图是每对顶点都相邻接的简单图。如果完全图有n个顶点，就记为K_n。可以看出，K_n有$n(n-1)/2$条边。作为例子，图K_5示于图1.12。

图G的所有顶点都具有相等度数时，图G叫做正则图。如果G是正则的，且G中每个顶点v_i的度数$d(v_i)=r$，那么G就叫做r-正则的。图1.13示出4-正则图。可以知道，K_n是$(n-1)$-正则图。

图$G=(V,E)$是二分的，如果它的顶点集合V可以划分为两个子集V_1和V_2，以使得E中每条边的一个端点在V_1中，另一个端点在V_2中，则(V_1,V_2)叫做G的二划分。如果在具有二划分的简单二分图G中，若V_1有m个顶点，V_2有n个顶点，对于V_1中每个顶点v_i和V_2中每个顶点v_j都存在一条边(v_i,v_j)，那么G就叫做完全二分图，记为$K_{m,n}$。

二分图和完全二分图$K_{3,4}$示于图1.14。

在图$G=(V,E)$中，如果能把V划分为k个子集V_1，V_2，……，V_k，并使得G的每条边的一个顶点在某个V_i中，而另一个顶点在某个V_j中$(i\neq j)$，则图G叫做k-分图。完全k-分图G是一个简单k-分图，它具有顶点集划分$(V_1, V_2, \cdots\cdots, V_k)$，而且图中

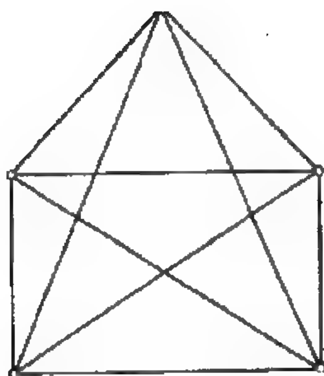


图 1.12 图K_5

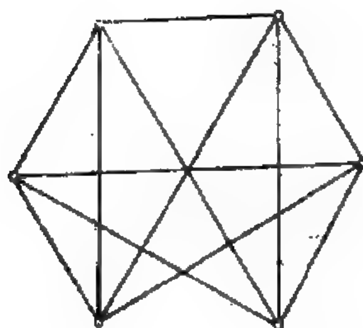


图 1.13 4-正则图

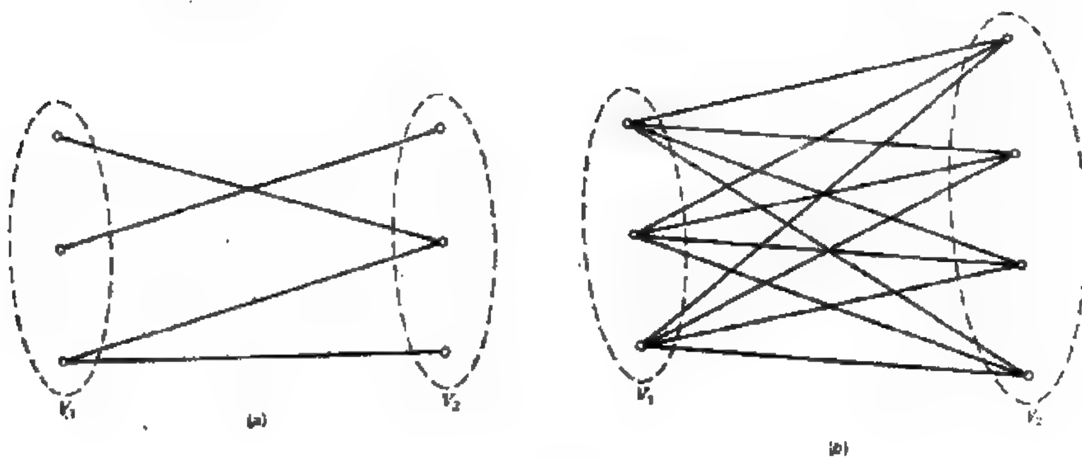


图 1.14
(a)二分图；(b)完全二分图$K_{3,4}$

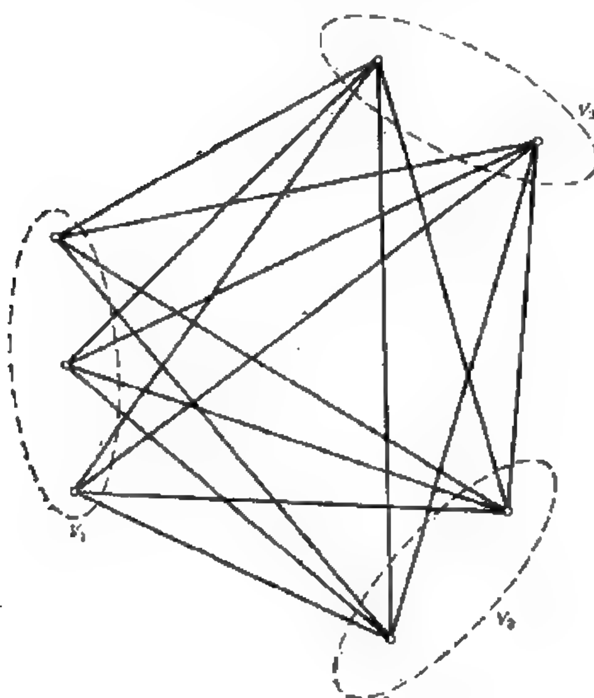


图 1.15 完全3-分图

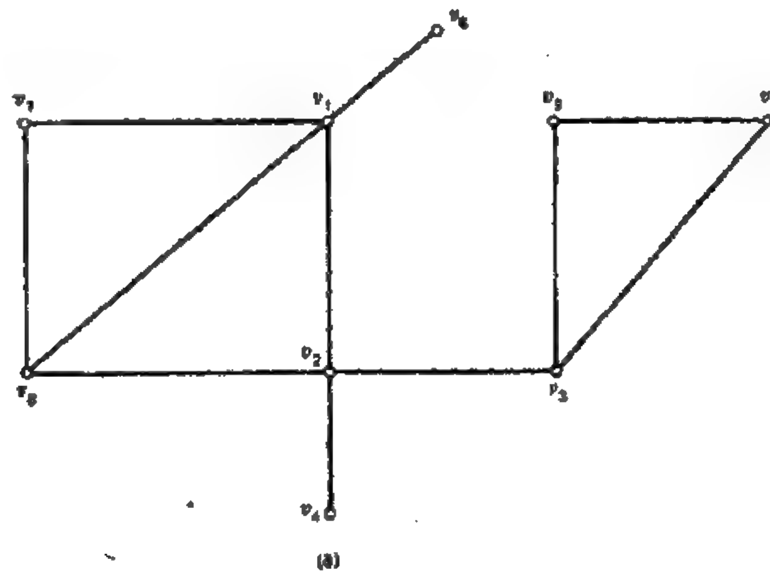
每个V_r的顶点v_i和V_s中的顶点v_j（$r\neq s$，$1\leqslant r$，$s\leqslant k$）之间都是G的一条边（v_i，v_j）。完全3-分图示于图1.15。

1.7 割点和可分图

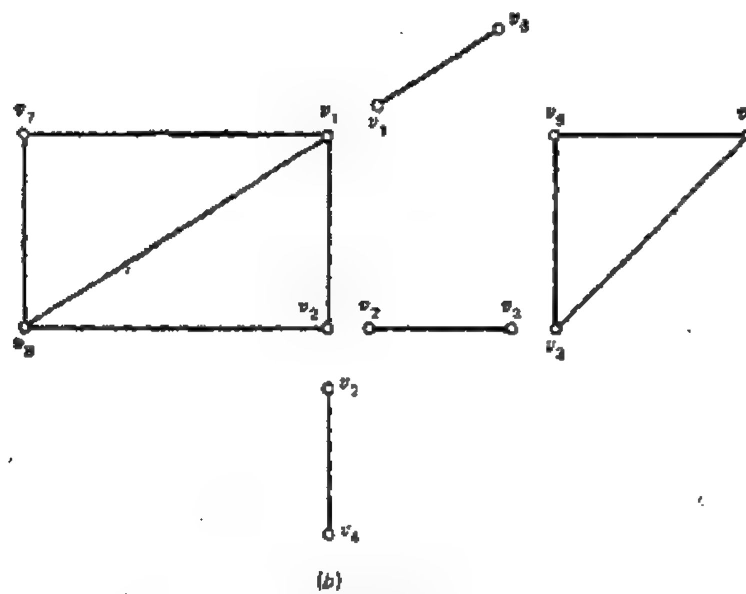
图G有顶点v_i，如果图$G-v_i$由比G更多数目的片组成，则v_i是G的割点．如果G是连通的，那么$G-v_i$至少包含两个片，即$G-v_i$不再是连通的．根据这个定义，孤立顶点不是割点。

如果一个图仅有一个顶点则称这个图是平凡的．因而平凡图没有割点。

不可分图是不含有割点的连通图．其它所有的图都是可分的．（注意不连通图是可分的）。



(a)



(b)

图 1.16 可分图及其块
(a)可分图G；(b)G的块。

图1.16(a)所示图G是可分的，它有三个割点v_1、v_2和v_3。

可分图G的块是G的最大不可分子图。图1.16(a)所示可分图G的块，示于图1.16(b)中。

下面的定理给出了割点的等价定义。

定理1.5. 顶点v是连通图G的割点，当且仅当存在两个不同于v的顶点u和w，并使得v位于每条$u—w$路径上。

证明

必要性 因为v是G的割点，由定义可知，$G—v$是不连通的。令G_1是$G—v$的一个片，V_1是G_1的顶点集，V_2是V_1在$V—v$中的补集。

u和w是两个顶点，u在V_1中，w在V_2中。考察G中任何一条$u—w$路径，如果割点v不位于这条路径上，那么这条路径仍在$G—v$中，即u和w在$G—v$中是连通的。然而，这与u和w是$G—v$中不同片相矛盾。于是，顶点v位于每条$u—w$路径上。

充分性 如果v位于每条$u—w$路径上，那么顶点u和w在$G—v$中是不连通的，于是，图$G—v$是不连通的。因而，由定义可知，v是割点。 □

仿照定理1.5，可给出下述可分图的等价定义：

连通图G是可分的，当且仅当G中存在顶点v，并使得它是两个非平凡真子图G_1和G_2的唯一公共顶点，G_1和G_2的并等于G。

假定图G是可分的，那么G的所有顶点都可能是割点吗？正如下面定理要证明的那样，回答是否定的。

定理1.6. 每个非平凡连通图至少包含两个非割点的顶点。

证明

我们通过对图的顶点数的归纳来证明这个定理。

定理对于每一个具有两个顶点的连通图是正确的，因为这些顶点中没有割点。

假设定理对所有含有少于n个顶点的非平凡连通图是正确的，其中$n>2$。令G是含有n个顶点的连通图。如果G不含割点，定理即可得证。

否则，令v是G的割点，且G_1，G_2…G_k是图$G—v$的k个片。如果任何G_j是平凡的，那么它唯一的顶点就不是割点。

考虑任何非平凡片G_j。由归纳假设可知，G_j包含了不是G_j割点的顶点v_1和v_2。显然，在这两个顶点中，如果某一个不与G中v相邻，那么这个顶点就不是G的割点。另一方面，如果v_1和v_2在G中与v相邻，那么v_1和v_2就都不是G的割点。

于是，$G—v$的每个片至少含有一个不是G的割点的顶点。因此，图G中至少有两个不是割点的顶点。 □

1.8 同构和2-同构

图1.17示出的两个图“看起来”是不同的。然而，一个图只要重画，再看上去就和另一个图完全一样。所以，这两个图在一定意义上是“等价的”。这种等价将在下面严格叙述。

两个图称为同构的，如果它们的顶点集之间和边集之间都存在一一对应关系，并且G_1和G_2的对应边与G_1和G_2的对应顶点相关联。即，如果G_1的顶点v_1和v_2分别对应于G_2的顶点v_1'和v_2'，那么G_1中以v_1和v_2为端点的边，一定对应于G_2中以v_1'和v_2'为端点的边，反之亦然。

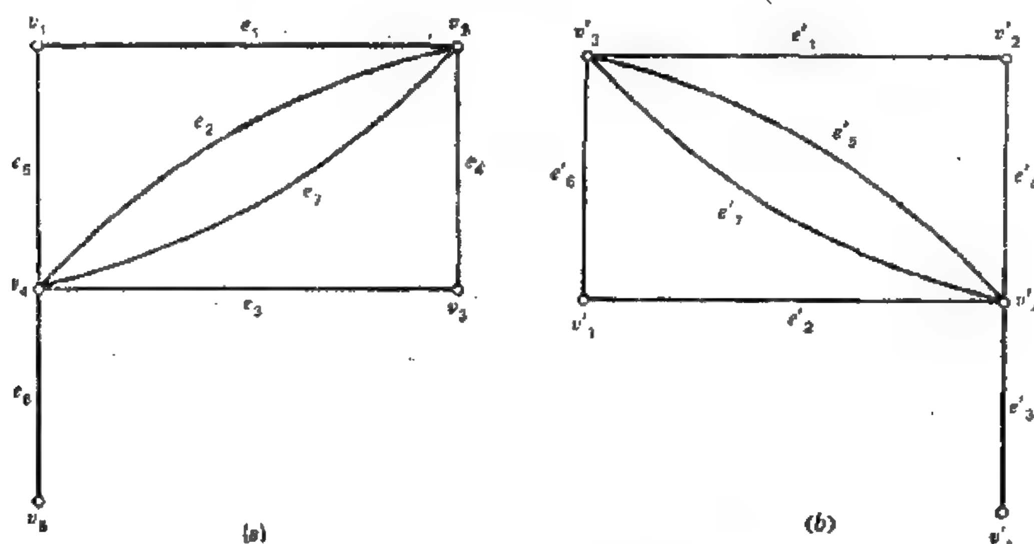


图 1.17 同构图
(a)图G_1；(b)图G_2。

根据上述定义，图1.17的两个图是同构的。它们的顶点集和边集分别对应如下：

顶点对应

$$v_1 \longleftrightarrow v_2',\ \ v_2 \longleftrightarrow v_3',\ \ v_3 \longleftrightarrow v_1',$$
$$v_4 \longleftrightarrow v_4',\ \ v_5 \longleftrightarrow v_5'.$$

边对应

$$e_1 \longleftrightarrow e_1',\ \ e_2 \longleftrightarrow e_5',\ \ e_3 \longleftrightarrow e_2',$$
$$e_4 \longleftrightarrow e_6',\ \ e_5 \longleftrightarrow e_4',\ \ e_6 \longleftrightarrow e_3',$$
$$e_7 \longleftrightarrow e_7'.$$

其次，考察图1.18(a)和(b)所示的两个可分图。这两个图不是同构的。假如我们把图G_1的割点v_1"分裂"成两个顶点，便得到图1.18(c)所示的两个边不相交图*。如果我们在图G_2的割点v_1'上进行相同的"分裂"操作，就得到图1.18(d)所示的两个边不相交图。可以看出，图1.18(c)和(d)是同构的。于是，两个图G_1和G_2在分裂顶点后就成为同构的。这样的图叫做1-同构的。

下面定义的2-同构是一种更加普遍类型的同构。两个图G_1和G_2是2-同构的，如果反复应用下面的一种或两种运算，使它们变为同构的。

1. 把G_1和（或）G_2中的割点"分裂"成两个顶点，便得到两个边不相交图。

2. 如果某一个图，比如说G_1，含有两个恰好有两个公共顶点v_1和v_2的子图G_1'和G_1''，

---

* 如果两个图没有公共边，则它们是边不相交的；如果两个图没有公共顶点，则它们是顶点不相交的。

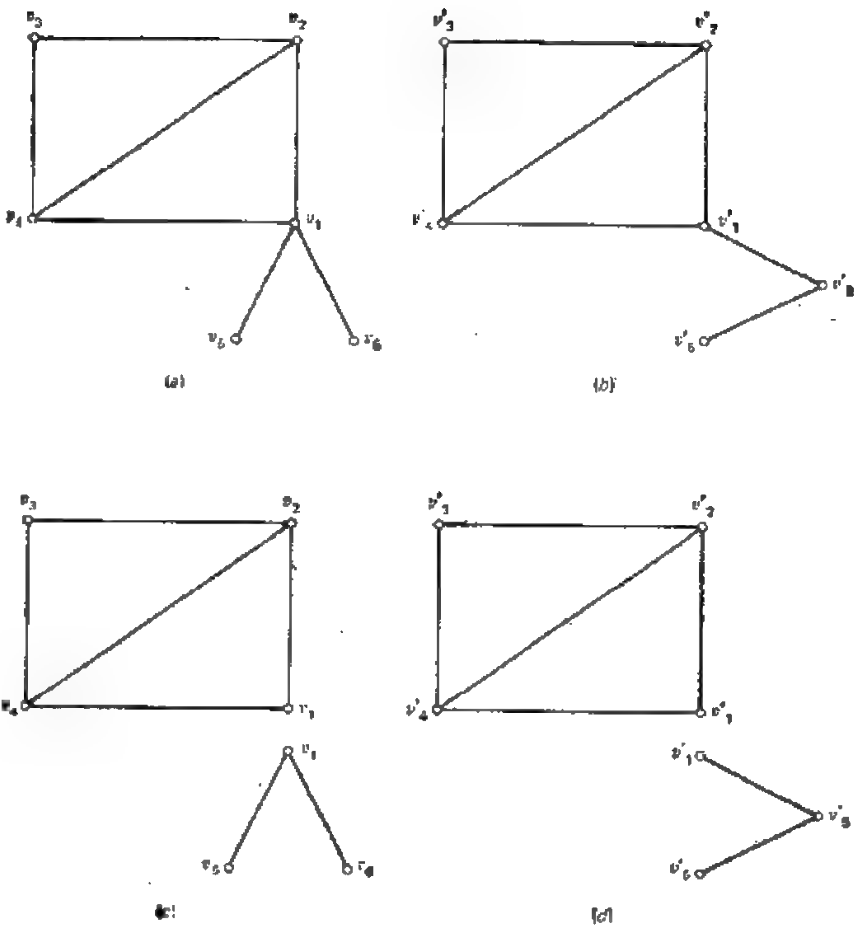


图 1.18 1-同构图

(a)G_1; (b)G_1; (c)分裂G_1中v_1的图; (d)分裂G_1中的图。

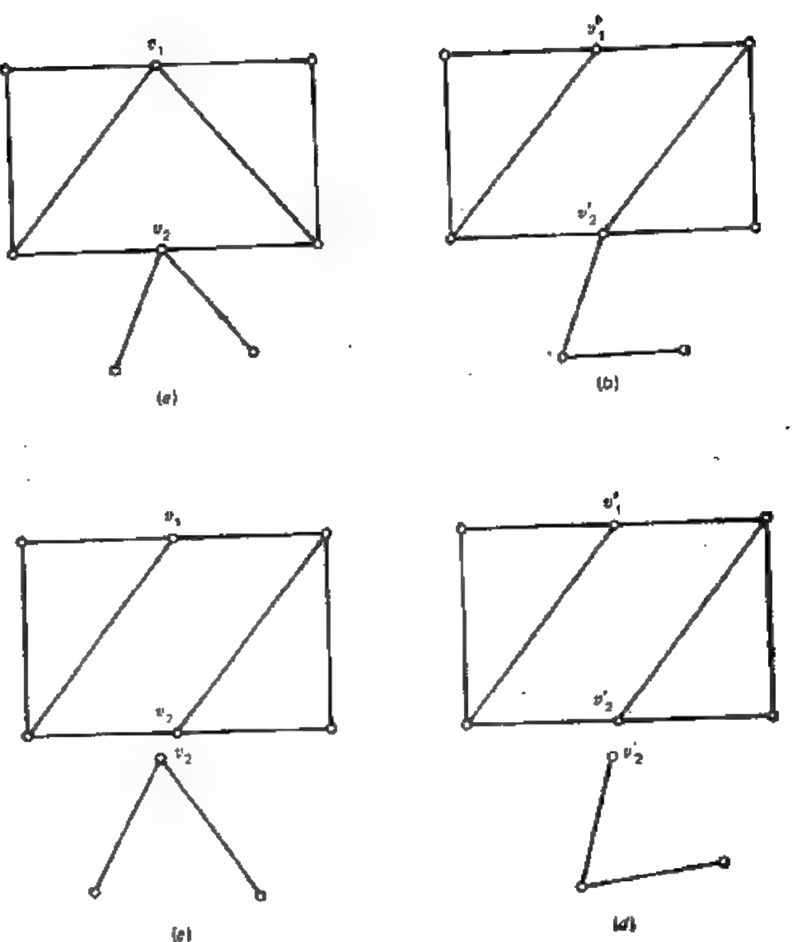


图 1.19 2-同构的解释

(a)G_1; (b)G_2; (c)G'_1; (d)G'_2。

那么就在某一子图中交换这两个顶点的名称。（几何上，这种运算等价于以公共顶点v_1和v_2为轴，“翻转”子图G'_1和G''_1中某一个。）

考察图1.19(a)和(b)所示图G_1和G_2。在G_1中对顶点v_3进行“分裂”操作，并在顶点v_1和v_2上进行“翻转”操作，就得到图1.19(c)所示图G'_1。在G_2中对顶点v'_1进行“分裂”操作，便得到图1.19(d)所示图G'_2。G'_1和G'_2这两个图是同构的，于是G_1和G_2是2-同构的。

下面的定理给出了2-同构图的重要结论。

定理1.7. 两个图G_1和G_2是2-同构的，当且仅当它们的边集存在一一对应关系，并使得一个图的回路对应于另一个图的回路。 □

当G_1和G_2是2-同构的，显然G_1的回路对应于G_2的回路。然而，这个定理的逆定理的证明相当冗长，所以不在此讨论。关于2-同构图的原始文献[1.1]讨论了这个问题。

1.9 进一步阅读

文献[1.2]和[1.3]是本书第一篇中所涉及到的几个专题的极好参考文献。文献[1.2]还讨论了超图和拟阵。文献[1.4]是一本文笔流畅的教科书，其中收集了许多图论上尚未解决的问题。文献[1.5]对图论作了精彩的初步介绍，还有一章讨论拟阵。其它关于图论的教科书有文献[1.6]，[1.7]和[1.8]。还有一些讨论到如计数、算法等专题的书和论文也可作为参考，这些将在相应章中提及。

1.10 习 题

1.1 图G有n个顶点和m条边，且每个顶点的度数是k或$k+1$。求证：若图G有n_k个顶点的度数是k，n_{k+1}个顶点的度数是$k+1$，则$n_k=(k+1)n-2m$。

1.2 证明下列说法正确与否：

(a) 任何两条连接两个顶点的不同闭通道的并包含了一个回路。

(b) 任何两条连接两个顶点的不同路径的并包含了一个回路。

1.3 在图G中，如果任意两个顶点a和b，以及任意两个顶点b和c之间均存在一条路径，那么证明：a和c之间必存在一条路径。

1.4 设P_1和P_2是图中任意两个顶点之间两条不同路径。证明：$P_1 \oplus P_2$是回路或边不相交回路的并。

1.5 证明：一条所有顶点度数都是2的闭合轨迹是一个回路。

1.6 证明：如果在图G中有两条不同回路包含了边e，那么G中必存在一个不包含边e的回路。

1.7 证明：对于$\delta(G)\geqslant k$的简单图G，必存在一条长度至少为k的路径。再证明必存在一个长度至少为$k+1$的回路。假定$k\geqslant 2$。

1.8 证明：图$G=(V, E)$是连通的，当且仅对于V的每一个划分(V_1, V_2)，V_1和V_2为非空集，G中必存在一条边连接V_1中的一个顶点和V_2中的一个顶点。

1.9 证明：具有n个顶点和k个片的简单图至多有$(n-k)(n-k+1)/2$条边。从这个结论可以推断出，如果G的边超过$(n-1)(n-2)/2$条，则它一定是连通的。

1.10 证明：如果图G（连通或非连通的）恰好有两个奇度数顶点，则一定存在一条连接这两个顶点的路径。

1.11 证明：如果简单图G是非连通的，则它的补图$\overline{G}$是连通的。

1.12 如果G是具有n个顶点、m条边的图，$m<n-1$。证明：G是非连通的。

1.13 如果图G有n个顶点、m条边，$m\geqslant n$，那么证明：G包含一条回路。

1.14 证明：具有n个顶点的简单图，当$\delta(G)\geqslant(n-1)/2$时，才是连通的。

1.15 证明：至少具有两个顶点的简单图包含两个相同度数顶点。

1.16 证明：如果图$G=(V, E)$是简单图，并且是连通的，但不是完全的。那么G必存在三个顶点u、v、w，使得边（u, v）和边（v, w）在E中，而边（u, w）不在E中。

1.17 如果顶点u和v在G中是连通的，则u和v间的距离就是最短u—v路径长度，记为$d(u, v)$。如果u、v间不存在路径，那么就定义$d(u, v)$为无穷大。G的直径是G中两个顶点间距离的最大值。证明：如果G的直径大于3，那么$\overline{G}$的直径就小于3。

1.18 图G的围长是G中最短回路的长度。如果G中无回路，就定义G的围长为无限大。证明：围长为4的k-正则图，至少有$2k$个顶点。

1.19 简单图G是自补的，如果它与其外图$\overline{G}$同构。证明：自补图的顶点数一定是$4k$或$4k+1$的形式，k是整数。

1.20 证明：n个顶点的简单图，当它的边数超过$n^2/4$，就是不可二分的。

1.21 证明图是二分的，当且仅当它的所有回路长度是偶数。

1.22 构造具有$2n$（$n\geqslant3$）个顶点、无三角形的简单立方图。

注：3-正则图又叫做立方图，三角形是长度为3的回路。

1.23 证明：如果v是简单图G的割点，那么v就不会是$\overline{G}$的割点。

1.24 证明：具有顶点数$n\geqslant3$的图G的下列性质是等价的。

(a) G是不可分的。

(b) G的每两个顶点在同一回路上。

(c) 对于G中每个顶点v和每条边e，总存在一个包含v和e的回路。

(d) G的每两条边在同一回路上。

(e) 给定G的两个顶点和一条边，必有一条连接这两个顶点，并包含这条边的路径。

(f) 对于G的每三个不同顶点，必存在一条连接任意两个顶点，并通过第三个顶点的路径。

(g) 对于G的每三个不同顶点，存在一条连接任意两个顶点、并不通过第三个顶点的路径。

1.25 证明：如果G没有偶数长度的回路，那么G的每个块不是K_1，就是K_2或奇数长度的回路。

1.26 证明：连通图若不是一个块，就是包含具有一个相同顶点的两个或更多的块。用$b(v)$表示图$G=(V, E)$中包含顶点v的块的数目。证明：图G中块的数目等于$p+\sum_{v\in V}(b(v)-1)$，p是G中片的数目。

1.27 令$c(B)$表示B块中连通图G的割点数。那么G中割点数$c(G)$由下式给出

$$c(G)-1=\sum_{\text{全部块}}[c(B)-1]$$

1.28 图G的桥是边e，并使得$G-e$比G有更多的片。证明下列性质：

(a) 连通图G的边e是桥，当且仅当存在两个顶点v、w，使得e在每条v—w路径上。

(b) 图G的边是G的桥，当且仅当它不在回路上。

1.29 图1.20所示图是否同构？为什么？

1.30 证明：图1.21的两个图不同构。

1.31 确定所有阶数为3和4的非同构简单图。

注：三个顶点的非同构图有四个，四个顶点的非同构图有十一个。

1.32 证明：任意两个具有n个顶点，每个顶点度数均为2的简单连通图是同构的。

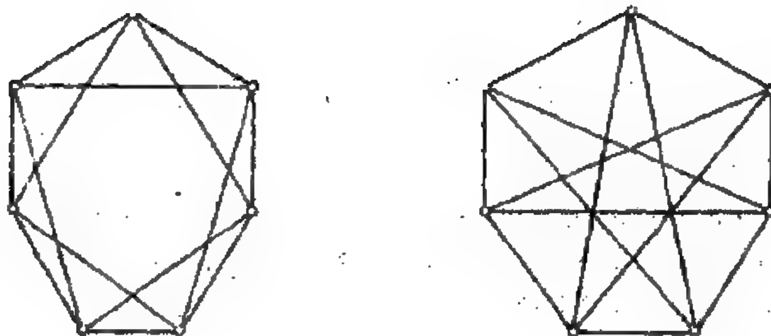


图 1.20

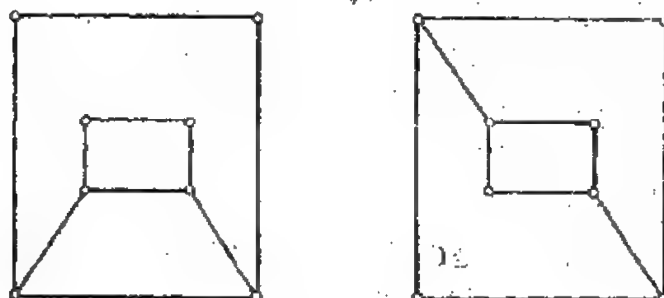


图 1.21

1.11 参考文献

1.1 H.Whitney, "2-Isomorphic Graphs," *Am.J.Math.*, Vol.55, 245-254(1933).

1.2 C.Berge, *Graphs and Hypergraphs*, North Holland, Amsterdam, 1973.

1.3 F.Harary, *Graph Theory*, Addison-Wesley, Reading, Mass.1969.

1.4 J.A.Bondy and U.S.R.Murty, *Graph Theory with Applications*, Macmillan, London, 1976.

1.5 R.J.Wilson, *Introduction to Graph Theory*, Oliver and Boyd, Edinburgh, 1972.

1.6 C.L.Liu, *Introduction to Combinatorial Mathematics*, McGraw-Hill, New York, 1968.

1.7 M.Behzad and G.Chartrand, *Introduction to the Theory of Graphs*, Allyn and Bacon, Boston, 1971.

1.8 N.Deo, *Graph Theory with Applications to Engineering and Computer Science*, Prentice-Hall, Englewood Cliffs, N.J., 1974.

第二章　树、割集和回路

在大部分应用中所遇到的图都是连通的。在连通图中，树具有最简单的结构，也许是最重要的结构。如果连通图是重要的，那么使连通图断开的边集就应具有同等的重要性。这样就引出了割集的概念。本章将研究树和割集以及几个与它们有关的结果，并且阐述它们与回路间的关系。

2.1　树、生成树和补生成树

如果一个图没有回路，那么就说它是无环的。树是连通无环图。图G的树是G的一个连通无环子图。图G的生成树是具有G的全部顶点的树。树T的连通子图叫做T的子树。

例如，考察图2.1（a）所示图G。图2.1（b）中的图G_1和G_2是G的两个树。图2.1(c)中的图G_3和G_4是G的两个生成树。

图G的生成树T的补生成树T^*是G的子图，它具有G的全部顶点以及那些没有包括在T中的所有边。注意，一个补生成树可以是不连通的。图2.1(c)中的生成树G_3和G_4的补生成树G_3^*和G_4^*，由图2.1(d)所示。

生成树T的边叫做T的树支，补生成树T^*的边叫做连支或弦。

生成树T唯一地确定了它的补生成树T^*。因此，我们把T^*的边称为T的弦或连支。

现在，我们开始讨论树的几个性质。

把树定义为连通无环图，虽然在概念上是简单的，但还存在表示树特性的其他几种等价方法。下面的定理将讨论这些方法。

定理2.1　对具有n个顶点、m条边的图G来说，下面的论点是等价的。

1. G是一个树。
2. G的任意两个顶点之间恰好存在一条路径。
3. G是连通的，且$m=n-1$。
4. G是无环的，且$m=n-1$。
5. G是无环的，如果G的任意两个非相邻的顶点由一条边连接起来，那么所产生的图恰好具有一个回路。

证明

$1\Longrightarrow 2$　见习题1.2b。

$2\Longrightarrow 3$　首先注意到，因为在G的任意两个顶点之间存在一条路径，所以G是连通的。

我们通过对G的顶点数进行归纳来证明$m=n-1$。对于具有一个或两个顶点的连通图，这是显然的。假定对于小于n个顶点的连通图，这也是正确的。

考虑G中的任何边e。边e构成了e的两端点间唯一的路径。因此在图$G-e$中，e

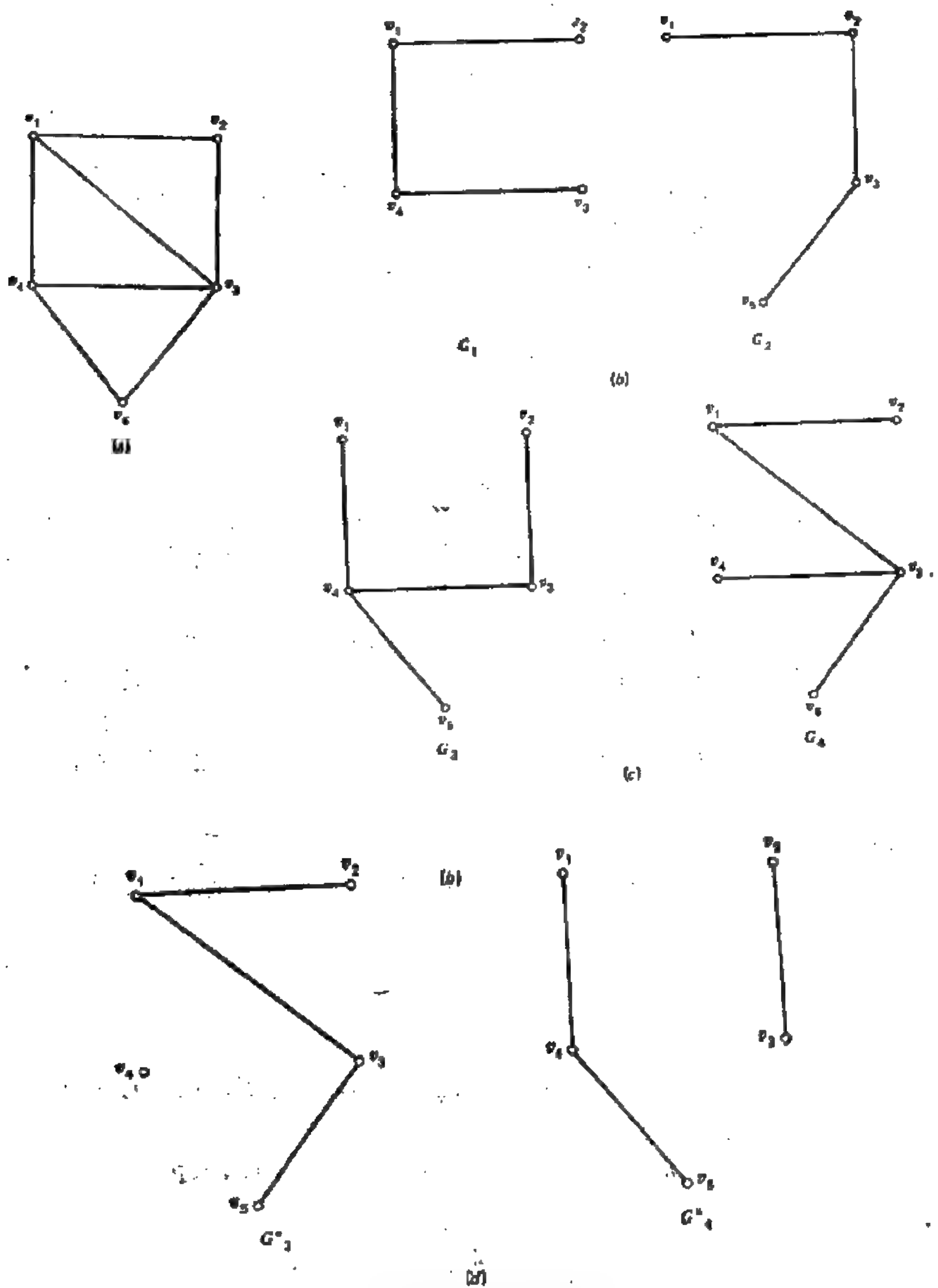


图 2.1 树，生成树和补生成树

(a)图G；(b)G的树G_1和G_2；(c)G的生成树G_3和G_4；(d)G 的补生成树G_3^* 和 G_4^*。

的端点间就没有路径。这样图$G-e$就是不连通的。进而图$G-e$应该恰好包含两个连通片，否则图G将是不连通的。

令G_1和G_2是$G-e$的两个连通片。令n_1和m_1分别是G_1中顶点数和边数。同样，令n_2和m_2分别是G_2中顶点数和边数。因此，我们有

$$n=n_1+n_2$$

和

$$m=m_1+m_2+1$$

注意G_1和G_2满足论点2的假设，即G_1和G_2中的任意两个顶点间恰好存在一条路径。因为$n_1<n$和$n_2<n$，由归纳法假设，有

$$m_1=n_1-1$$

和

$$m_2=n_2-1$$

因此

$$m=n_1-1+n_2-1+1=n-1$$

$3\Longrightarrow4$　令图G重新命名为G_0，即$G_0=G$。

假定G_0有某些回路。然后设从G_0中移走一条回路边（比如e_1）后所产生的图为G_1，即$G_1=G_0-e$。因为G_0是连通的，根据定理1.4可得，G_1也是连通的，并且含有G_0的全部n个顶点，而且G_1的边数等于$m-1$。

如果G_1不是无环的，令e_2是G_1的一条回路边，则图$G_2=G_1-e_2=G_0-e_1-e_2$也应该是连通的，并含有G_1的全部n个顶点，而且G_2有$m-2$条边。如果G_2不是无环的，重复上述过程，直到得出一个无环的连通图G_p为止。注意，G_p有n个顶点和$m-p$条边。

因为G_p是连通和无环的，所以它必然是一个树。因而根据本定理的上述论点得出

$$m-p=n-1$$

因为由假设$m=n-1$，可得$p=0$。因此图$G=G_0$是无环的。

$4\Longrightarrow5$　设G_1，G_2，…，G_p是G的p个连通片，n_i和m_i分别表示连通片G_i中的顶点数和边数。那么

$$m=m_1+m_2+\cdots+m_p$$

和

$$n=n_1+n_2+\cdots+n_p$$

每个连通片G_i是连通的。因为G是无环的，所以G_i也是无环的。由此得出G_i是一个树。然后，根据本定理论点3，

$$m_i=n_i-1,\qquad \text{对于所有}\quad 1\leqslant i\leqslant p$$

因此得到

$$m=\sum_{i=1}^{p}m_i=\sum_{i=1}^{p}(n_i-1)=n-p$$

根据假设

$$m=n-1$$

从而得出

$$p=1$$

这样G恰好含有一个连通片，因此G是连通的。又因G是无环的，所以G是一个树。那么根据本定理论点2，在G的任何两个相异顶点之间恰好存在一条路。因此，如果对G加上一条边$e=(v_1, v_2)$，那么这条边与v_1和v_2间的唯一路径连起来所产生的图中就恰好形成一个回路。

$5\Longrightarrow 1$ 假设G是不连通的，考虑属于G的不同连通片中的任意两个顶点v_a和v_b，那么v_a和v_b在G中是不连通的。

因为G中v_a和v_b间没有路，所以在G上加一条边(v_a, v_b)也不会产生回路。但是，这与论点5相矛盾。因此G是不连通的假设不能成立。这样G是连通的。 □

因为G也是无环的，所以根据定义，它必然是一个树。

在上面定理论点的1到5的每一条中，都清楚地表达了图G是树的充要条件。此定理的直接推论如下。

推论2.1.1 考虑具有n个顶点图G的子图G'。设G'具有n个顶点和m'条边，那么下面的论点是等价的：

1.G'是G的生成树。

2.G'的任何两个顶点间恰好存在一条路径。

3.G'是连通的，且$m'=n-1$。

4.G'是无环的，且$m'=n-1$。

5.G'是无环的，如果G'的任何两个非相邻顶点被一条边连接起来，那么它所产生的图恰好具有一个回路。 □

在上述推论中没有包括但很容易证明的一个条件，叙述如下。

推论2.1.2 n个顶点图G的子图G'是G的一个生成树，当且仅当G'是无环的、连通的、且有$n-1$条边。 □

显而易见，n个顶点图G的一个子图具有下面性质中的任意三条，它就应是G的一个生成树：

1.它有n个顶点。　　2.它是连通的。

3.它有$n-1$条边。　　4.它是无环的。

因此会提出这样一个问题，为了确定一个生成树，满足上述四条性质中的任意两条是否为充分条件。下面将回答这个问题。（也可参见习题2.3）

定理2.2 n个顶点图G的一个子图G'是G的一个生成树，当且仅当G'是无环的，且有$n-1$条边。

证明

必要性可根据定理2.1论点4得出。

为了证明充分性，我们必须证明G'是连通的，且具有G的全部n个顶点。

设G'由p个连通片G_1, G_2, …, G_p组成，用n_i表示连通片G_i中的顶点数。令n'为G'中的顶点数，那么

$$n'=\sum_{i=1}^{p} n_i$$

每个G_i是连通的，因为G是无环的，则每个G_i也是无环的。这样每个G_i是一个树，且有n_i-1条边。因此G'中边的总数等于

$$\sum_{i=1}^{p}(n_i-1)=n'-p$$

但根据假设　　　　　　$n'-p=n-1$

因为$n'\leqslant n$和$p\geqslant 1$，显然上述方程是正确的，当且仅当$n'=n$和$p=1$。这样G'是连通的，且有n个顶点。因为G'也是无环的，由定义，G'是G的一个生成树。□

设图G有一棵生成树T，因为G的子图T是连通的，且有G的全部顶点，那么G也应该是连通的，其次，我们希望证明这个结论的逆，即一个连通图至少有一个生成树。

如果一个连通图G是无环的，那么它就是自身的生成树。如果不是，则令e_1是G的某个回路边，由定理1.4，图$G_1=G-e_1$仍是连通的，且有G的全部顶点。如果G_1不是无环的，重复上述过程，直到我们得到一个连通无环图G_p、且有G的全部顶点为止。那时图G_p就将是G的一个生成树。

下一个定理概括了上面讨论的结果。

定理2.3　图G是连通的，当且仅当它有一个生成树。□

因为图G的生成树T是无环的，所以每一个T的子图都是G的一个无环子图。那么是否G的每个无环子图都是G的某个生成树的子图？回答是肯定的，这在下一定理中将要证明。

定理2.4　连通图G的子图G'是G的某个生成树的一个子图，当且仅当G'是无环的。

证明

必要性是显然的。

为了证明充分性，令T为图G的一个生成树。考察图$G_1=T\cup G'$。显然，G'是G_1的一个子图。因为T是G_1的一个子图，则G_1是连通的，且有G的全部顶点。如果G_1是无环的，那么它就是一个生成树，G'就是G_1的一个子图，定理得证。（注意，如果G_1是无环的，且$G_1=T$，那么G'就是T的一个子图）。

假定G_1有一个回路C_1。因为G'是无环的，由此可知，C_1的全部边并不都在G'中。这样，C_1一定至少有一条边（譬如e_1）不在G'中。从G_1中移走这条回路边e_1，便产生了图$G_2=G_1-e_1$，G_2也是连通的，且含有G_1的全部顶点。注意，G'是G_2的一个子图。如果G_2是无环的，则G_2就是所要求的生成树。如果不是，重复这个过程，直到求得一个生成树，并使得G'是它的一个子图为止。□

下面我们证明一个关于悬挂顶点（即树中度数为1的顶点）最小数目的有趣定理。

定理2.5　在非平凡树中，至少存在两个悬挂顶点。

证明　假定一棵树T有n个顶点，那么根据定理2.1，它有$n-1$条边。由定理1.1，我们又有

$$\sum_{i=1}^{n} d(v_i)=2\times T\text{中的边数}$$

这样

$$d(v_1)+d(v_2)+\cdots+d(v_n)=2n-2$$

只有当上式左边项中至少有两项等于1，即T至少有两个悬挂顶点时，此方程才是正确的。 □

2.2 k-树、生成k-树和林

一个k-树是由k个连通片组成的无环图。显然，k-树的每个连通片本身就是一个树。注意，1-树和树是等同的。

如果一个k-树是图G的生成子图，那么它就叫做图G的生成k树。

图G的生成k-树T的补生成k-树T^*是G的生成子图，它恰好包含了那些在G中而不在T中的边。

例如，图2.2(b)的图就是图2.2(a)中所示图G的一个2-树。G的生成3-树T和所对应的补生成3-树T^*示于图2.2(c)和(d)。

设n个顶点图G的生成k-树的k个连通片用T_1，T_2，…，T_k表示。如果n_i是T_i中顶点数，那么

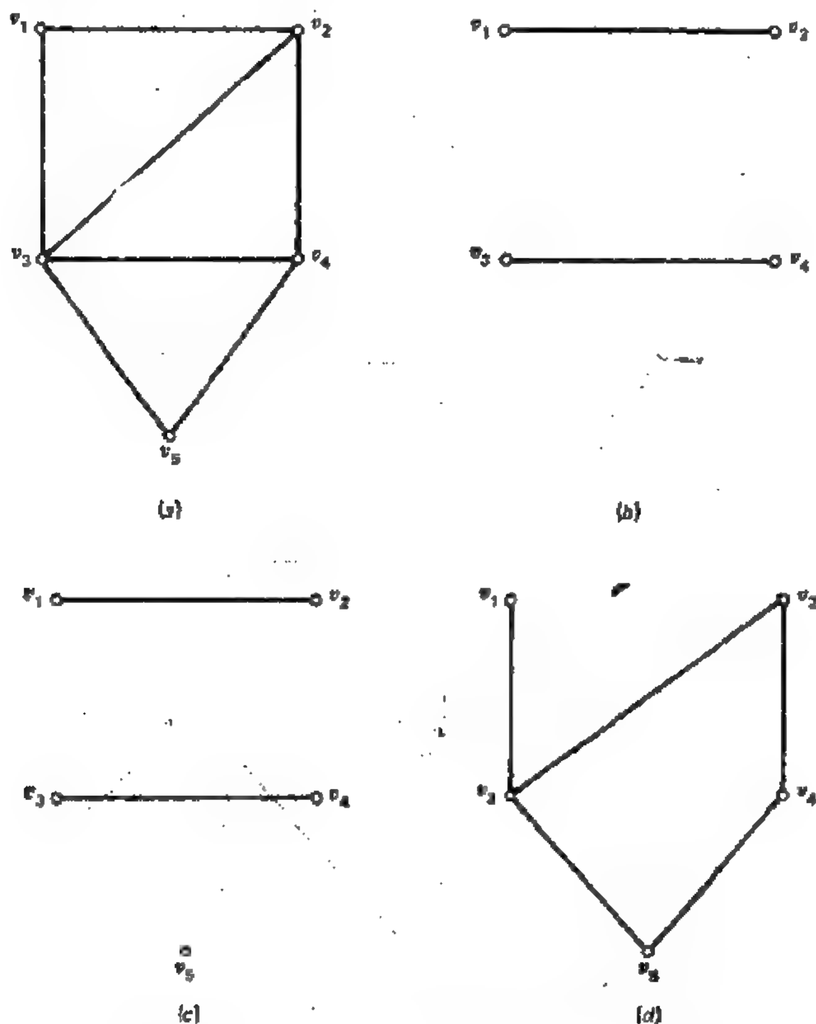


图 2.2 k-树，生成k-树和补生成k-树定义的说明

(a)图G；(b)G的一个2-树；(c)G的一个生成3-树T；(d)补生成3-树T^*。

$$n=n_1+n_2+\cdots+n_k$$

因为每个T_i是一个树，根据定理2.1，有

$$m_i=n_i-1$$

其中m_i是T_i中的边数。

这样，生成k树T中边的总数为

$$\sum_{i=1}^{k} m_i=\sum_{i=1}^{k}(n_i-1)=n-k$$

如果m是G的边数，那么补生成树T^*将有$m-n+k$条边。

图G的一个林是G的一个生成k-树，这里k是G的连通片数。

如果图G有P个连通片，那么对G的任何生成k-树，则$k\geqslant P$。因为G的一个林T是$k=P$时G的一个生成k-树，则T的每一部分必然是G的某个连通片的一个生成树。这样，有P个连通片G_1，G_2，…，G_p的图G的一个林T由P个连通片T_1，T_2，…，T_p组成，其中T_i是G_i的一个生成树，$1\leqslant i\leqslant p$。

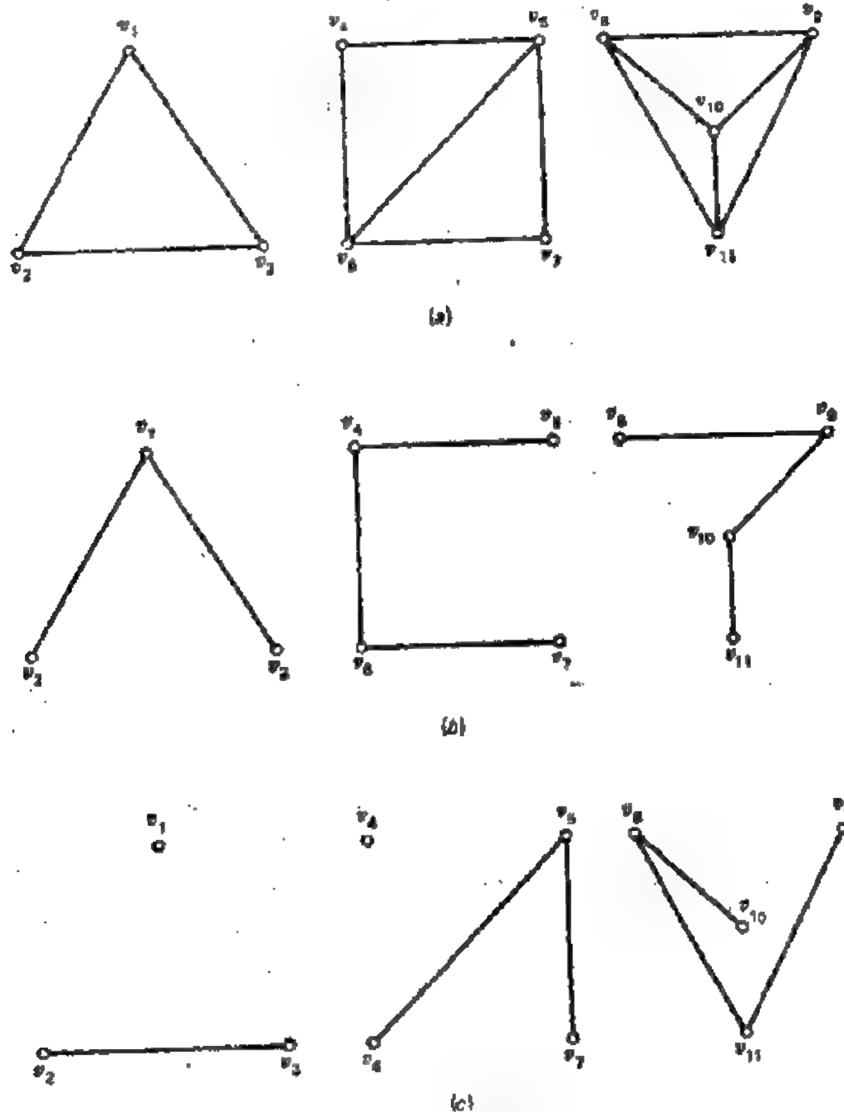


图 2.8 林和补林

(a)图G；(b)G的林T；(c)补林T^*。

图G的一个林T的补林T^*是G的生成子图，它恰好包含了那些不在T中的边。

注意，在一个连通图内，林和生成树是同义语。

一个图的林T和对应的补林T^*示于图2.3。

2.3 秩和零度

设有m条边，n个顶点和k个连通片的图G。G的秩定义为等于$n-k$，记为$\rho(G)$；G的零度定义为等于$m-n+k$，记为$\mu(G)$。注意

$$\rho(G)+\mu(G)=m$$

由林和补林的定义可知，图G的秩$\rho(G)$等于G的林的边数，而G的零度$\mu(G)$等于G的补林的边数。

数$\rho(G)$和$\mu(G)$是些与图有关的最重要的数目。在第四章将看到，$\rho(G)$和$\mu(G)$定义了图的割集和回路子空间的维数。

2.4 基本回路

考察连通图G的一个生成树T。令T的树枝用b_1，b_2，…，b_{n-1}表示，T的连支用

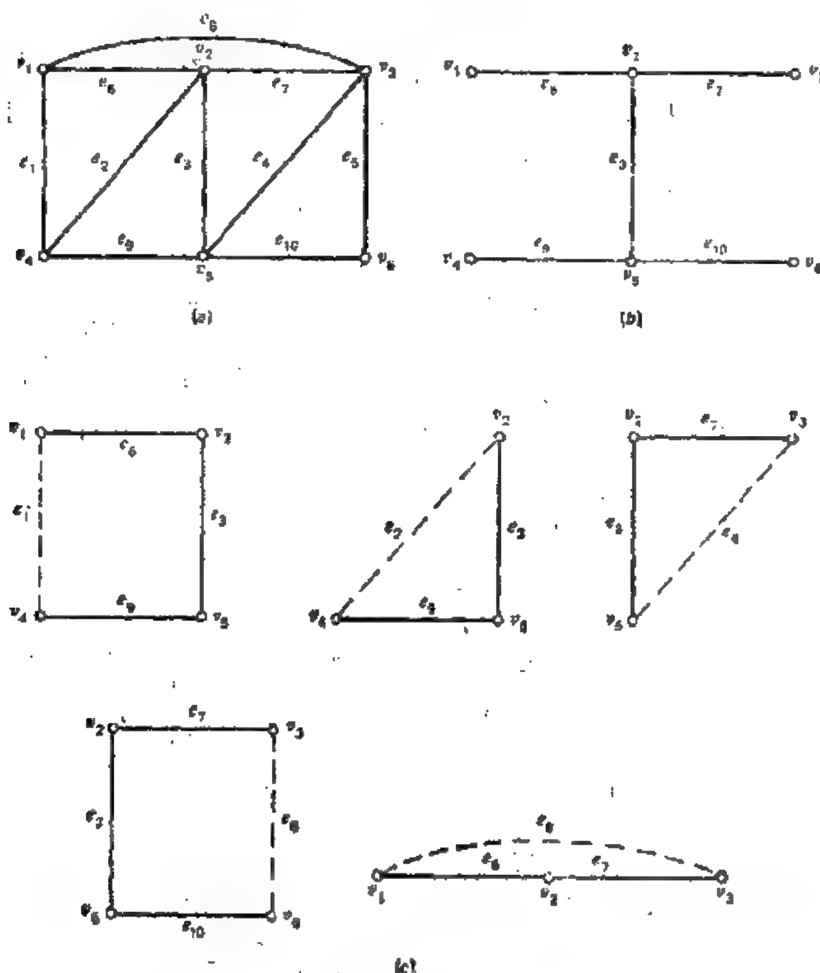


图 2.4 图G的基本回路集

(a)图G；(b)G的生成树T；(c)相对于T的G的5个基本回路(弦用破折号表示)。

c_1，c_2，…，c_{m-n+1}表示，其中m是G的边数，n是G的顶点数。

T是无环的，根据定理2.1，图$T\cup c_i$恰好包含一个回路C_i。这个回路由弦c_i和T中的一些边组成，其中T的那些边位于连接c_i端点间的唯一路径上。此回路C_i叫做G相对于生成树T的弦c_i的基本回路。

相对于生成树T的弦而言，G的全部$m-n+1$个基本回路C_1，C_2，…C_{m-n+1}的集称为图G相对于T的基本回路集。

基本回路C_i的一个重要特征是，它恰好包含了一个弦，即弦c_i。此外，弦c_i不出现在相对于T的其它基本回路中。由于这些性质，没有一个基本回路的边集能够表示为某些或全部其他的基本回路边集的环和。在第四章将证明，图G的每个回路都能够表示为对G的某一生成树而言的G的某些基本回路的环和。正由于这些原因，“基本”回路才这样称呼。

图G和G的基本回路集示于图2.4。

2.5 割集

连通图G的割集S是G的一个最小边集，若从G中移走S，G就不连通，即图$G-S$是不连通的。

例如，考察图2.5(a)所示图G的边子集$S_1=\{e_1, e_3, e_7, e_{10}\}$。若从$G$中移走$S_1$，就产生了图2.5($b$)中的图$G_1=G-S_1$，$G_1$是不连通的。此外，移走$S_1$中任何真子集，不能使$G$不连通，这样$S_1$就是$G$的一个割集。

考察另一个集$S_2=\{e_1, e_2, e_5, e_7, e_9\}$。图2.5($c$)所示的图是不连通的。然而，

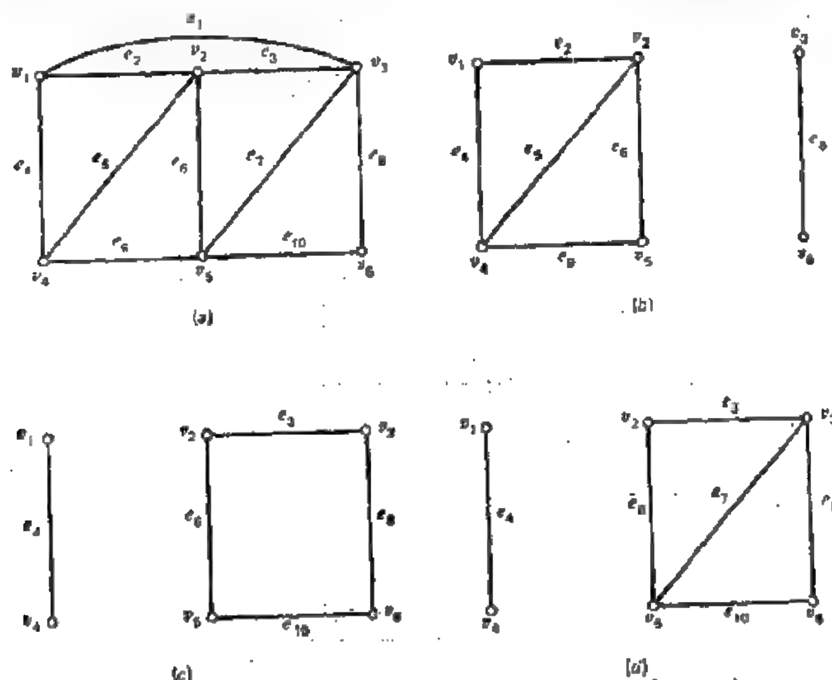


图 2.5 割集定义的说明

(a)图G；(b)$G_1=G-S_1$，$S_1=\{e_1, e_3, e_7, e_{10}\}$；

(c)$G_2=G-S_2$，$S_2=\{e_1, e_2, e_5, e_7, e_9\}$；($d$)$G_2'=G-S_2'$，$S_2'=\{e_1, e_2, e_5, e_9\}$。

集$S_2'=\{e_1, e_2, e_5, e_6\}$是$S_2$的一个真子集，也能使$G$不连通。图$G_3=G-S_2'$示于图2.5(*d*)。这样，$S_2$就不是$G$的一个割集。

注意，根据上面给出的割集定义，如果S是图G的一个割集，那么G的秩和$G-S$的秩至少相差1，即$\rho(G)-\rho(G-S)\geqslant 1$。

文献[2.1]定义的割集如下：

连通图G的一个割集S是G的一个最小边集，以致移去S就使G恰好分成两个连通片，即$\rho(G)-\rho(G-S)=1$。

现在要问，这两个割集定义是否是等价的。回答是"是"，它的证明留作练习（习题2.15）。

2.6 切　割

现在我们定义切割的概念，它与割集的概念密切相关。

考察一个有顶点集V的连通图G。令V_1和V_2为两个互相分开的V的子集，并使得$V=V_1\cup V_2$；即V_1和V_2没有公共顶点，且合起来包含了V的所有顶点。于是，G中那些一个端点在V_1中，另一个端点在V_2中的所有边构成的集S叫做G的一个切割，通常记为$\langle V_1, V_2\rangle$。文献[2.2]把切割称为seg（使顶点集V分开的边集）。

例如，对于图2.6所示图G，如果$V_1=\{v_1, v_2, v_3, v_4\}$及$V_2=\{v_5, v_6, v_7\}$，那么$G$的切割$\langle V_1, V_2\rangle$等于边集$\{e_6, e_7, e_8\}$。

注意，G的切割$\langle V_1, V_2\rangle$是G的最小边集，若移去切割$\langle V_1, V_2\rangle$就把G分为两个图G_1和G_2，则图G_1和G_2是在顶点集V_1和V_2上的G的导出子图。G_1和G_2可能是不连通的。

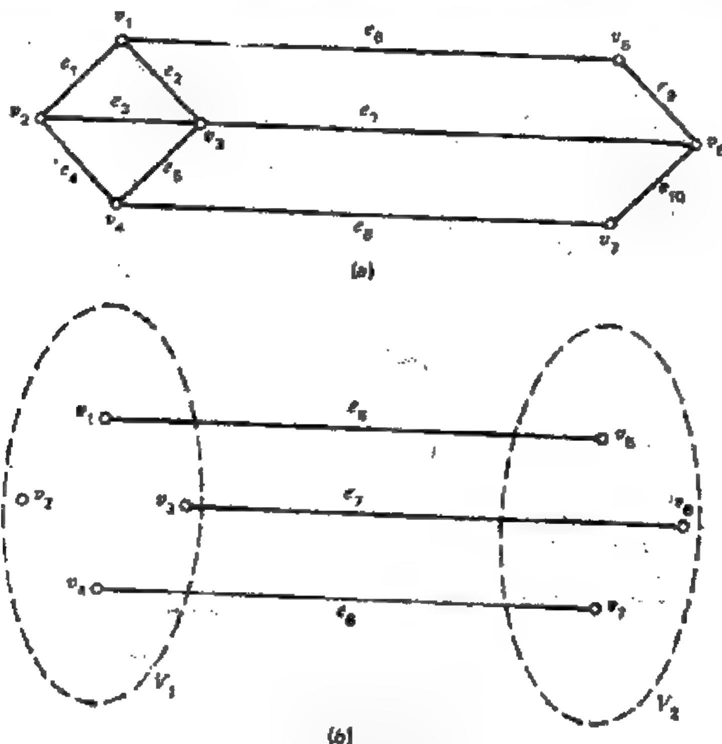


图 2.6 切割的定义

(*a*)图G；(*b*)G的切割$\langle V_1, V_2\rangle$。

如果图G_1和G_2都是连通的，那么$\langle V_1, V_2\rangle$也就是恰好把G分成两个连通片的最小边集。于是，根据定义，$\langle V_1, V_2\rangle$是G的一个割集。

假定对于G的一个割集S，V_1和V_2分别是$G-S$的两个连通片G_1和G_2的顶点集，那么S就是切割$\langle V_1, V_2\rangle$。

这样，我们有了下面的定理。

定理2.6 1.若连通图G在其顶点集V_1和V_2上的导出子图是连通的，则G的切割$\langle V_1, V_2\rangle$是G的一个割集。

2.若S是连通图G的一个割集，V_1和V_2是$G-S$的两个连通片的顶点集，那么$S=\langle V_1, V_2\rangle$。 □

在连通图G中，任何切割$\langle V_1, V_2\rangle$包含了G的一个割集，因为从G中移走$\langle V_1, V_2\rangle$，将使G不连通。事实上，我们能够证明，图G的一个切割和G的某些边不相交割集的并是一致的。在下面的定理中将阐述这一点。

定理2.7 连通图G的一个切割是G的某些边不相交的割集的并。 □

这个定理证明不难，留作练习（习题2.10）。

下面考察连通图中的顶点v_1。与v_1关联的边集形成了切割$\langle v_1, V-v_1\rangle$。若移去这些边，就将把$G$分为两个子图，所以由定义可知，这两个子图中仅含顶点v_1的子图是连通的，而另一个子图是在顶点集$V-v_1$上的G的导出子图G'。因此，切割$\langle v_1, V-v_1\rangle$就是一个割集，当且仅当$G'$是连通的。然而，$G'$是连通的，当且仅当$v_1$不是割点（1.7节）。这样我们有下面定理。

定理2.8 连通图G中与顶点v关联的边集是G的一个割集，当且仅当v不是G的一个割点。 □

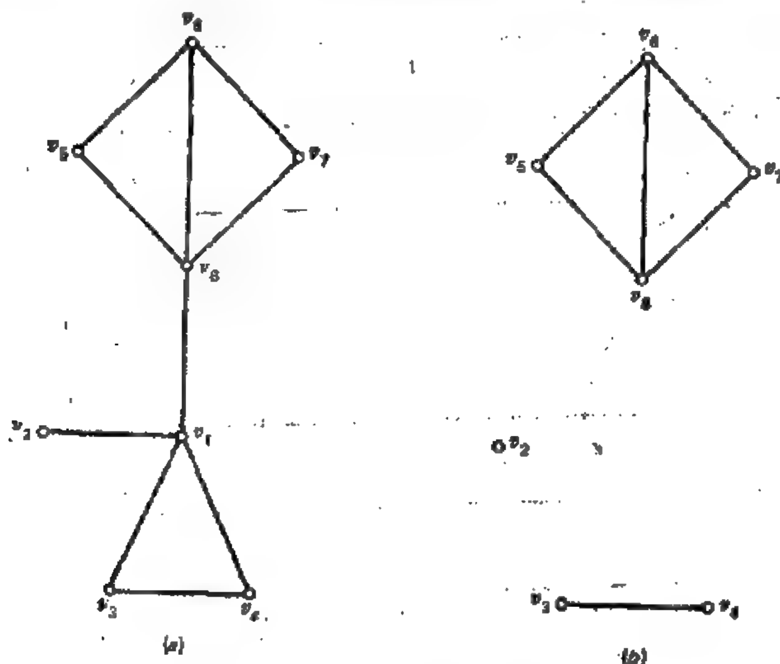


图 2.7 定理2.8的说明

(a)图G；(b)顶点集{v_2, v_3, v_4, v_5, v_6, v_7, v_8}上G的导出子图。

例如，考察图2.7(a)所示的可分图G。v_1是G的一个割点。在顶点集$V-v_1=\{v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$上$G$的导出子图示于图2.7($b$)。这个子图由三个连通片组成，但不是连通的。这样，与割点v_1关联的边集就形不成G的一个割集。

2.7 基本割集

在2.4节中已经说明了，如何利用连通图的一棵生成树来求得图的基本回路集。在本节，我们将说明如何利用一个生成树来确定基本割集的集合。

考察连通图G的一个生成树T。令b为T的一个树支。若移走b，将使T恰好分为两个连

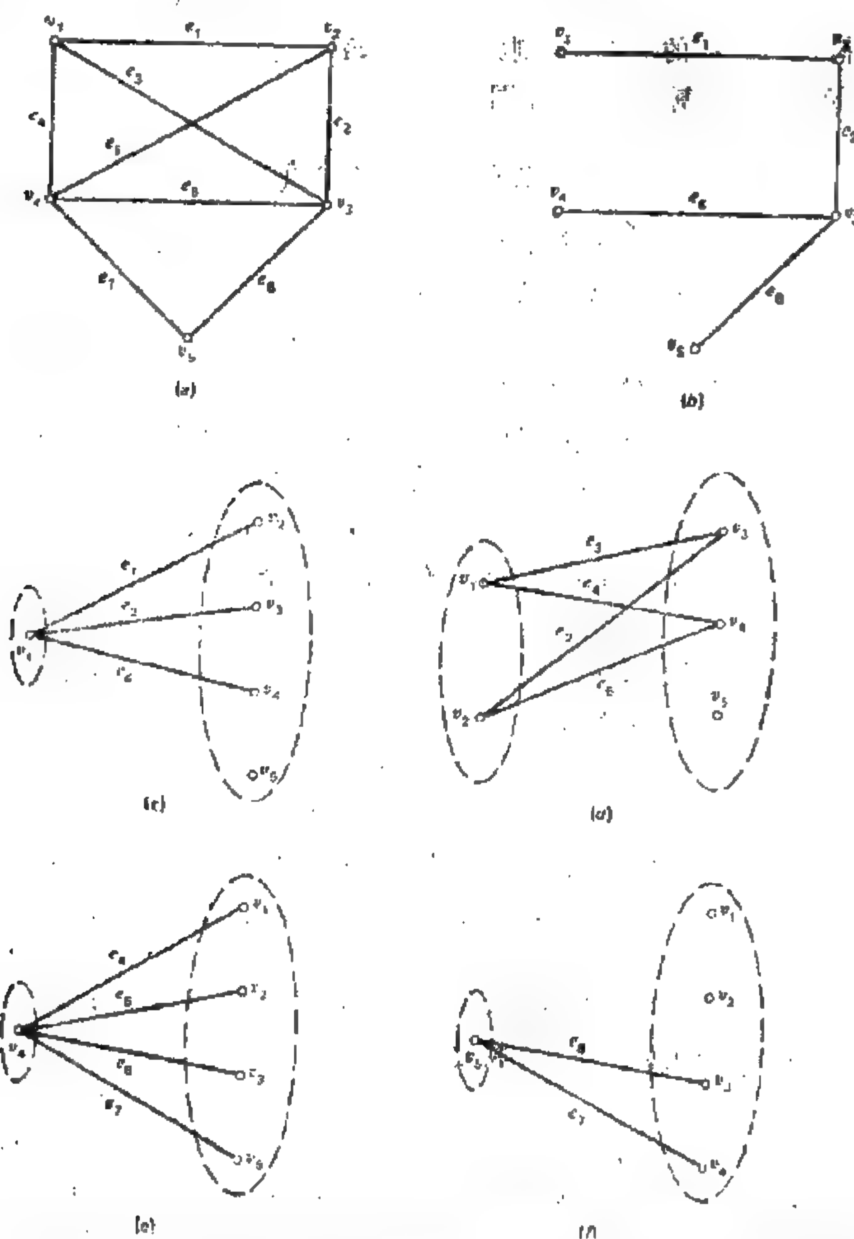


图 2.8 图的基本割集集合(a)图G；(b)G的生成树；(c)相对于树支e_1的基本割集；(d)相对于树支e_2的基本割集；(e)相对于树支e_6的基本割集；(f)相对于树支e_8的基本割集。

通片T_1和T_2。注意，T_1和T_2是G的两个树。令V_1和V_2分别表示T_1和T_2的顶点集，V_1和V_2合起来包含了G的所有顶点。

设G_1和G_2分别是顶点集V_1和V_2上G的导出子图。显而易见，T_1和T_2分别是G_1和G_2的生成树。因此，根据定理2.3，G_1和G_2是连通的。这再次证明了（定理2.6）切割$\langle V_1, V_2\rangle$是G的一个割集。这个割集称为相对于G的生成树T中树支b而言的G的基本割集。相对于G的生成树T的$n-1$个树支而言的全部$n-1$个基本割集称为基本割集集合。

注意，割集$\langle V_1, V_2\rangle$恰好包含了一个树支，即T的树枝b；而$\langle V_1, V_2\rangle$的其他所有边都是T的连支，这可由$\langle V_1, V_2\rangle$不包含T_1或T_2的任何边这个事实得到。此外，树支b不出现在相对于T的任何其他基本割集中。由于这些性质，没有一个基本割集的边集能够用某些或全部其他基本割集的环和来表示。在第四章中，将证明图G的任一割集都能够用相对于G的生成树T而言的G的某些基本割集的环和来表示。

图G以及它的基本割集集合示于图2.8。

2.8 生成树、回路和割集

本节将讨论分别与生成树和补生成树相联系的割集和回路的一些有趣结果。这些结果将显示出回路和割集的对偶性质。也将推出另一种分别用生成树和补生成树来描述割集和回路的性质的方法。

很显然，从连通图G中移走割集S，这就破坏了G的所有生成树。稍加思索就能理解，割集是一个最小边集，若从G中移走这个边集就破坏了G的所有生成树。但是这个结果的逆并非如此明显。本节的最初几个定理将讨论这些问题，以及与回路有关的类似问题。

定理2.9 连通图G的一个割集，至少包含G的每个生成树的一个树支。

证明

假定G的割集S不包含G的某个生成树T的树支，那么图$G-S$将包含生成树T。因此，根据定理2.3，$G-S$是连通的。但这与S'是G的割集相矛盾。 □

定理2.10 连通图G的一个回路，至少包含G的每个补生成树的一条边。

证明

假定G的回路C不包含G的生成树T的补生成树T^*的边。那么图$G-T^*$将包含回路C。因为$G-T^*$和生成树T是一样的，这样就意味着生成树T包含一个回路。但这与生成树的定义相矛盾。 □

定理2.11 连通图G的一个边集S是G的一个割集，当且仅当S是个最小边集，此边集S至少包含G的每个生成树的一个树支。

证明

必要性：如果G的边集S是G的一个割集，那么由定理2.9可知，S至少包含G的每个生成树的一个树支。若S不是这样的最小边集，则S的真子集S'将包含G的每个生成树的一个树支。那么$G-S'$将不包含G的生成树，它将是不连通的。因此，移走G的割集S的一个真子集，将使G不连通。但这与割集的定义相矛盾。因此必要性得证。

充分性：如果S是至少包含G的每个生成树的一个树支的最小边集，那么图$G-S$将不包含生成树，因此它将是不连通的。假定S不是一个割集，那么S的真子集S'将是一个割集。因此，根据本定理的必要性部分，S'将是至少包含G的每个生成树的一个树支的最小边集。但这与S是至少包含G的每个生成树的一个树支的最小边集相矛盾。 □

上面的定理是依据生成树给出了割集的特征。下面我们希望建立依据补生成树给出回路的类似的特征。

考察图G中组成一个回路的边集C。由定理2.10，C至少包括了G的每个补生成树的一条边。现在我们证明，C不会有真子集C'具有这个性质。

很明显，C'不包含回路。因此由定理2.4，我们能构造一个包含C'的生成树T。对应于T的补生成树T^*与C'没有公共边。因此，对于C的每个真子集C'存在至少一个与C'没有公共边的补生成树T^*。实际上，这个论述对图的每一个无环子图也是正确的。这样给出下面的定理。

定理2.12 连通图G的回路是G的一个最小边集，该边集至少包含G的每个补生成树的一条边。 □

下边是上面定理的逆。

定理2.13 若C是至少包含了G的每个补生成树的一条边的最小边集，则连通图G的边集C就是G的一个回路。

证明

如前所示，边集C不能是无环的，这是因为对于每个G的无环子图G'，都存在一个与G'没有任何公共边的补生成树。这样，C至少有一个回路C'。假定C'是C的一个真子集，那么由定理2.12，C'是至少包含了G的每个补生成树的一条边的最小边集。但这与C是这样的最小边集的假设相矛盾。因此，不存在C的真子集是回路。因为C不是无环的，所以C必定是回路。

定理2.12和2.13确立了连通图G的边集C是一个回路，当且仅当它是至少包含G的每个补生成树的一条边的最小边集。

由定理2.11、2.12以及2.13给出的割集和回路的新特征，清楚地显示出割集和回路概念的对偶性。这种对偶性在第十章中将作进一步探讨，在那里将讨论拟阵理论。

下一个定理表明了割集和回路的关系但不涉及到树。

定理2.14 连通图的一个回路和一个割集之间有偶数个公共边。

证明

令C是连通图G的一个回路，S是G的一个割集。令V_1和V_2是$G-S$的两个连通子图G_1和G_2的顶点集。

如果C是G_1或G_2的一个子集，那么很明显，C和S的公共边数等于0，这是个偶数。

如果C和S有某些公共边。让我们从V_1集的一个顶点v_1出发，沿着回路C行走。因为这种行走应在v_1结束，这就需要我们每遇到从V_1中顶点到达V_2中顶点的S的一条边，就一定有S的一条边使我们从V_2中顶点回到V_1中顶点。这只有当C和S有一个偶数公共边才可能的。

上述定理是非常重要的，它形成了割集和回路间正交关系的基础。这个关系将在第四章讨论。

我们想指出，定理2.14的逆并不是十分正确的。但在第四章我们将证明，图G的一个边集S是一个割集（回路）或某些边不相交割集（回路）的并，当且仅当S与每个回路（割集）有偶数个公共边。

连通图的基本回路和基本割集是相对于图的生成树来定义的。因此毫不奇怪，基本回路与基本割集本身就是有关联的。正如下面定理所证。

定理2.15

1. 相对于连通图生成树T的一个连支的基本回路，恰好由包含了这个连支的基本割集中那些T的树支所构成。

2. 相对于连通图生成树T的一个树支的基本割集，恰好由包含了这个树支的基本回路中那些T的连支所构成。

证明

1. 令C为相对于连通图G的生成树T的连支c_1的基本回路。除了c_1以外，设C包含T的树支b_1，b_2，…，b_k。

假定S_i是相对于G的T的树支b_i的基本割集，其中$1 \leqslant i \leqslant k$。树支$b_i$是$C$和$S_i$共有的唯一树支。连支$c_1$是$C$中的唯一连支。因为$C$和$S_i$必须有一个偶数的公共边，这就需要基本割集$S_i$包含$c_1$。其次我们将证明$T$的其他基本割集都不包含$c_1$。

假定相对于T的某个树支b_{k+1}的基本割集S_{k+1}包含c_1。那么c_1将是C和S_{k+1}间唯一的公共边。这将与定理2.14相矛盾。这样，弦c_1仅仅出现在由树支b_1，b_2，…，b_k所确定的那些割集中。

2. 这部分的证明与部分1的证明类似。 □

2.9 进一步阅读

在有关图的回路和割集的若干结果的发展中，树的概念是中心。连通图的生成树、回路和割集恰巧对应地是拟阵的基、回路和补回路，而拟阵可用图的边集来定义。因此本章的结果对理解第十章中拟阵理论的发展是有帮助的。

电网络理论是图论最早应用的领域之一。本章的结果及第四章和六章的一些结果构成了图论对电网络研究的基础。Seshu和Reed[2.1]的开拓性工作以及Kim和Chien[2.3]，Chen[2.4]，与Mayeda[2.5]等教材被极力推荐为本课题进一步阅读材料。

有关树的若干问题在文献中已被广泛地研究了。有些问题在本书后面章节还会讨论到。

2.10 习　题

2.1 证明：在六个顶点上恰好有六个非同构树，七个顶点上有十一个树。画出这些树。

2.2 证明：树是一个二分图。

2.3 考虑n个顶点的连通图G的子图G_s。证明：除(b)、(d)一对条件外，下列条件中再没有其它条件

时可以说明G_t是G的生成树：

（a）G_t含有n个顶点。

（b）G_t含有$n-1$条边。

（c）G_t是连通的。

（d）G_t不包含回路。

2.4 证明：连通图G中每条悬挂边（与悬挂顶点相关联的边）包含在G的每个生成树中。

2.5 证明：连通图G的每条边是G的某个生成树的树支。

2.6 证明：树中每个度数大于1的顶点是割点。

2.7 证明：不可分图G的每条边可以成为G的某个补生成树的连支。

2.8 证明或否定：一个不可分图的任意两条边能够包含在某个基本回路上。

2.9 在什么条件下，图G的任意两条边能够成为G的某个补生成树的连支。

2.10 证明：不可分图的零度为1，当且仅当它是一个回路。

2.11 证明：任何一个图的零度都是非负的，给出一个零度为0的图例。

2.12 考虑在图G上进行下列两种运算：

（a）如果仅有两条边$e_1=(v_1, v_2)$和$e_2=(v_3, v_2)$与顶点v_2关联，那么就用一条连接v_1和v_3的边代替e_1和e_2。

（b）用两条边（v_1, v_0）和（v_2, v_0）代替（v_1, v_2），其中v_0是一个G中的不在新顶点。

证明：在上面的运算下，G的零度不变。

2.13 如果对于连通图G的每条边e，图$G-e$是不连通的，那么连通图G就是最小连通的。证明：一个连通图是一个树，当且仅当它是最小连通的。

2.14 证明：连通图G的子图G_t是G的生成树，当且仅当它是G的不包含回路的最大子图。

2.15 证明：连通图G的割集是G的最小边集S，移去S，就可使G成为两个片，即$\rho(G)-\rho(G-S)=1$。

2.16 证明：每个连通图包含一个割集。

2.17 证明：连通图G的子图可以包括在G的补生成树中，当而仅当它不包含G的割集。

2.18 证明：连通图G的一个边子集S构成了G的一个补生成树，当且仅当它是不包含G的割集的最大边子集。

2.19 证明：连通图G的切割是G的一个割集或一些边不相交割集的并。

2.20 证明：对于有两个以上顶点的不可分图，它的每个割集至少包含两条边。

2.21 证明：图G是不可分的，当且仅当它的每两条边在同一割集上。

2.22 令C是图G中的一个回路。a和b是C中任意两条边。证明：割集S的存在，使得$S\cap C=\{a, b\}$。

2.23 令T_1、T_2是连通图G的生成树。证明：如果e是T_1中任一条边，那么T_2中必存在一条边f，使得$(T_1-e)\cup f$（在T_1中用边f来代替e所得到的新图）也是G的一个生成树。再证明：每次用T_2中的一条边来代替T_1中的一条边，只要每步所得到的图是G的生成树，那么就可以把T_1变换成T_2。

2.24 （a）令C_1和C_2是图G的两个回路，边e_1既在C_1又在C_2中，边e_2在C_1中但不在C_2中。证明：存在回路C_3，而$C_3\subseteq(C_1\cup C_2)-e_1$，且$e_2$在回路$C_3$中。

（b）用割集代替回路，重复做（a）。

注：这个结果是Whitney在文献[2.6]中定义拟阵的"回路"时用过的假定之一（第十章）。

2.25 令T是在$k+1$个顶点上的任意一个树。证明：如果G是简单的且$\delta(G)\geq k$，那么G有一个与T同构的子图。

2.26 证明：图G包含k个边不相交的生成树，当且仅当对V的每一个划分（V_1，$V_2\cdots$，V_l），边的端点在不同划分上的边数至少为$k(l-1)$。（见文献[2.7]和[2.8]）

2.27 图$G=(V, E)$的中心是顶点u，从而使得$\max_{v\in V}\{d(u, v)\}$尽可能的小。其中$d(u, v)$是u和v间的距离。证明：一个树中恰好有一个中心或两个相邻接的中心。

2.28 具有n个顶点的连通图G的树图是顶点为G的生成树T_1，$T_2\cdots$，T_r的图，其中T_i与T_j是邻接的当且仅当它们恰好有$n-2$条共同边。证明：连通图的树图是连通的。（见文献[2.9]）。

提示：见习题2.23。

2.11 参考文献

2.1 S.Seshu and M.B.Reed, *Linear Graphs and Electrical Networks*, Addison-Wesley, Reading, Mass, 1961.

2.2 M.B.Reed, "The Seg: A New Class of Subgraphs" *IRE Trans Circuit Theory*, Vol.CT-8, 17—22(1961).

2.3 W.H.Kim and R.T.Chien, *Topological Analysis and Synthesis of Communication Notworks*, Columbia University Press, New York, 1962.

2.4 W.K.Chen, *Applied Graph Theory*: *Graphs and Electical Networks*, North-Holland, Amsterdam, 1971.

2.5 W.Mayeda, *Graph Theory*, Wiley-Intetscience, New York, 1972.

2.6 H.Whitney, "On the Abstact Properties of Linear Dependence," *Am.J.Math.*, Vol.57, 509—533(1935).

2.7 C.St.J.A.Nash-Williams, "Edge-Disjoint Spanning Trees of Finite Graphs," *J.London Miath.Soc*, Vol.36, 445--450(1961).

2.8 W.T.Tutte, "On the Problem of Decomposing a Graph into n Connected Factors," *J.London Math.Soc*, Vol.36, 221—230(1961).

2.9 R.L.Cummins, "Hamilton Circuits in Tree Graphs," *IEEE Trans.Circuit Theory*, Vol.CT-13, 82—90(1966).

第三章　欧拉图和哈密顿图

图论中的很多发现可追溯到试图解决若干"实际"问题——猜谜、游戏，等等。这些问题之一是著名的哥尼斯堡桥问题。这个问题可陈述如下：

在德国的哥尼斯堡Pregel河上有两个岛。这些岛相互之间以及与岸边之间由七座桥连接起来，如图3.1(a)所示。本问题是从四块陆地（图3.1(a)中用A、B、C、D标志）中任何一地出发，走过每座桥恰好一次，然后返回到出发点，即建立一个经过所有七座桥的闭合路径，但不重复经过其中的任何一座桥。

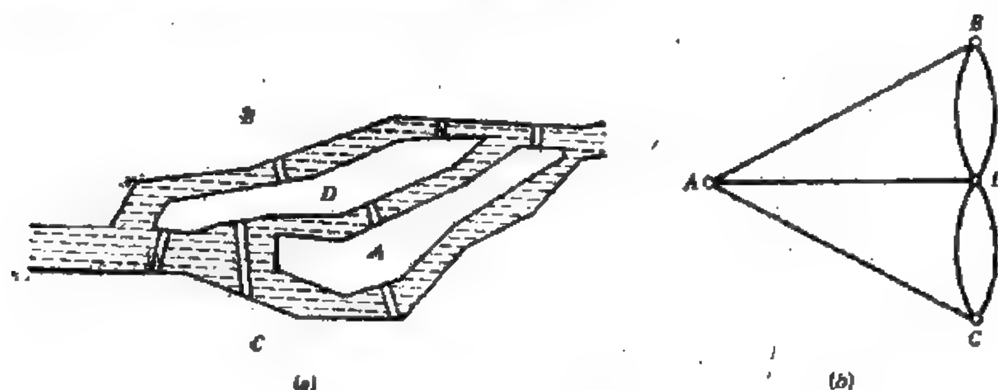


图 3.1　(a)哥尼斯堡桥问题；(b)哥尼斯堡桥问题的图。

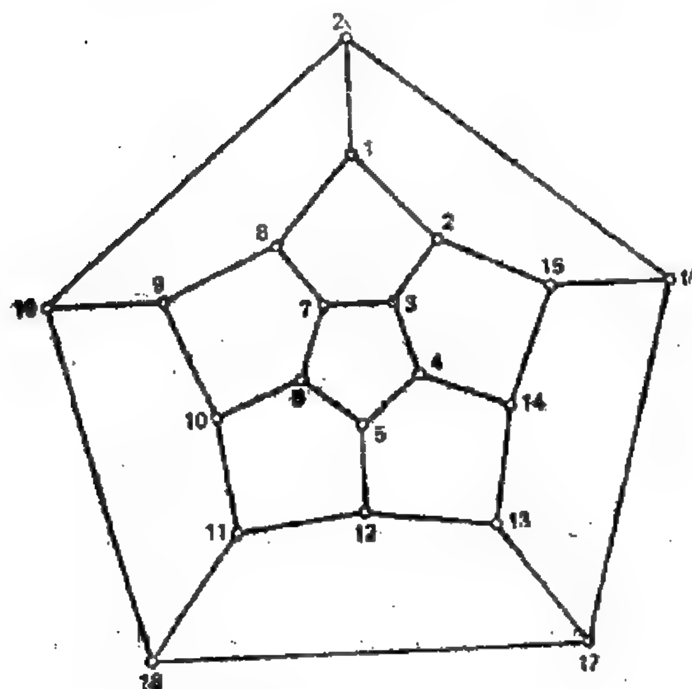


图 3.2　哈密顿游戏的图

很多人确信这个问题没有解。但伟大的瑞士数学家欧拉在1736年证明了这一点，奠定了图论的基础。欧拉首先证明了本问题与沿着图3.1(b)的边建立一条闭合轨迹是等价的，这里顶点A、B、C及D代表陆地，边代表连接陆地的桥。然后他推广了这个问题，建立了存在这样轨迹图的特性。这些图后来被称为欧拉图。3.1节的讨论涉及到这些图。

1859年，另一个伟大数学家哈密顿发明了一种游戏，它要求玩耍者沿着十二面体的边走出一条闭合回路，经过每个顶点恰好一次。用图论语言来讲，哈密顿游戏等价于确定在图3.2所示的十二面体图上能否找到一个生成回路，即包含所有20个顶点的回路。可以证明，顶点序列1、2、……、20、1构成了图3.2中的一个生成回路。凡含有生成回路的图后来都称为哈密顿图。3.2节的讨论涉及到这些图。

3.1 欧 拉 图

所谓图G中的欧拉轨迹是一条包含G的所有边的闭合轨迹。开欧拉轨迹是一条包含G的所有边的不闭合轨迹。含有欧拉轨迹的图就叫作欧拉图。

考虑图3.3(a)所示图G_1。边序列e_1、e_2、e_3、e_4、e_5、e_6、e_7、e_8、e_9、e_{10}、e_{11}和e_{12}构成了G_1中的一条欧拉轨迹。因此，G_1是欧拉图。

在图3.3(b)所示图G_2中，边序列e_1、e_2、e_3、e_4、e_5、e_6、e_7、e_8、e_9、e_{10}、e_{11}、e_{12}和e_{13}构成了一个开欧拉轨迹。但是，在G_2中没有欧拉轨迹，因此G_2不是欧拉图。

没有开欧拉轨迹的非欧拉图G_3，如图3.3(c)所示。

下面定理给出了欧拉图的简单而有用的特性。

定理3.1 下面论点对连通图G是等价的：

1. G是欧拉图。
2. G中每个顶点的度是偶数。
3. G是某些边不相交回路的并。

证明：

$1\Longrightarrow 2$ 令T是G中一条欧拉轨迹。假定我们从G中一个顶点，比如说从v_1出发，沿着T行走。令T为

$$v_1=x_1,\ e_1,\ x_2,\ e_2,\ x_3,\cdots,\ e_r,\ x_{r+1}=v_1$$

当然这里所有的边都是相异的；顶点x_2，…，x_r不可能都是相异的，这些顶点中的某一个可能是v_1。那么很显然，一对连续的边e_i和e_{i+1}对顶点x_{i+1}的度数贡献为2，$1\leqslant i\leqslant r-1$。再加上，初始边$e_1$和最后边$e_r$对顶点$v_1$的度数贡献为2。这样，所有顶点的度都是偶数。

$2\Longrightarrow 3$ 因为G是连通的，而G中每个顶点的度是偶数，则G中每个顶点的度大于1。这样G没有悬挂顶点，根据定理2.5，G因此不是一个树。这意味着G至少有一个回路，不妨说为C_1。

考虑图$G_1=G-C_1$。因为C_1中每个顶点的度也是偶数，由此可得G_1中每个顶点的度一定是偶数。但G_1可能是不连通的。

如果G_1是完全不连通的，即G_1仅仅包含孤立顶点，那么$G=C_1$，论点3得证。否则，

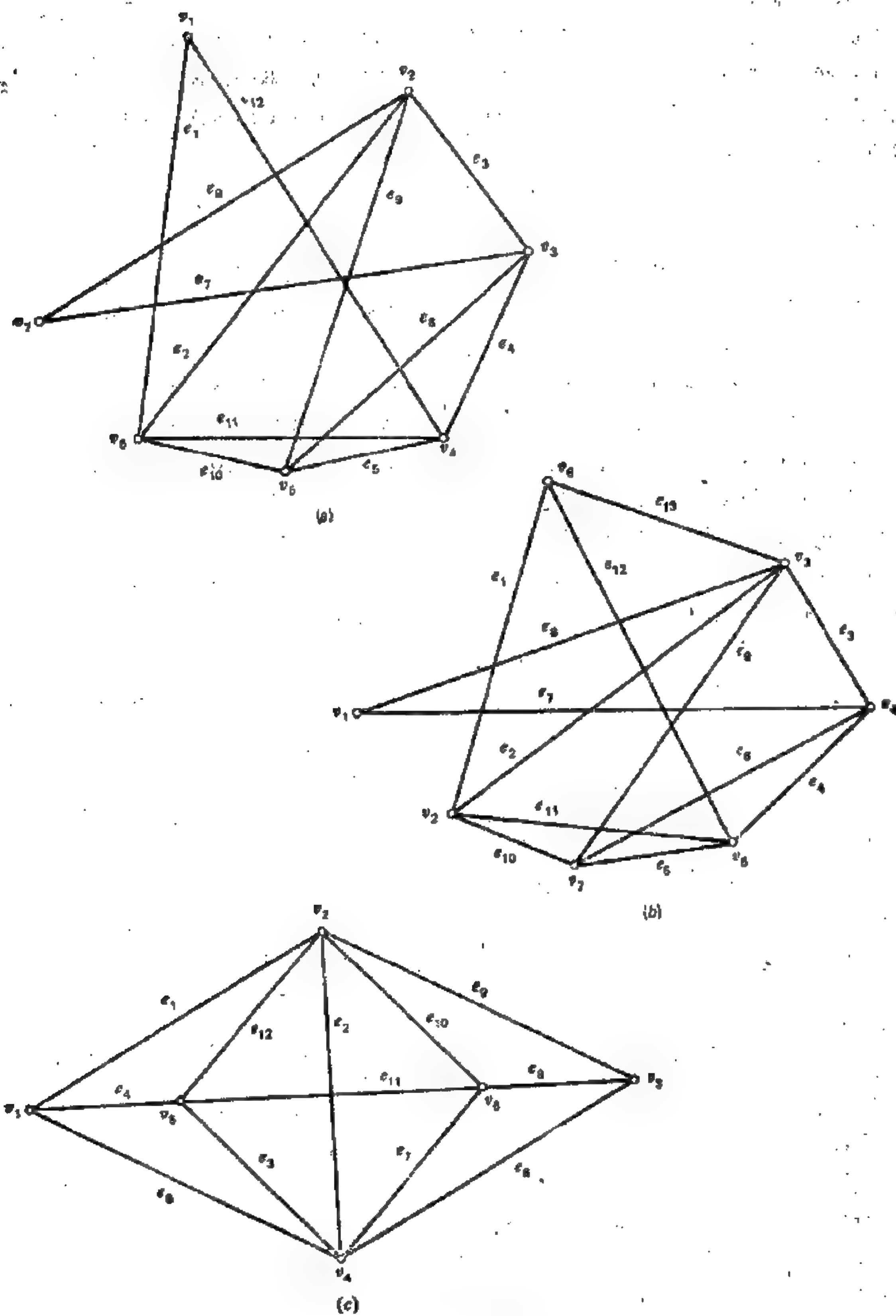


图 3.3 (a)G_1，一个欧拉图；(b)G_2，一个有开欧拉轨迹的非欧拉图；(c)G_3，一个没有开欧拉轨迹的非欧拉图。

G_1至少有一个回路C_2。

考虑下一个图$G_2=G_1-G_2=G-C_1-C_2$，G_2中每个顶点的度又是偶数。如果G_2是完全不连通的，那么$G_2=C_1\cup C_2$。否则重复这个过程，直到我们得到了一个完全不连通的图$G_n=G-C_1-C_2-\cdots-C_n$为止，这里C_1，C_2，…，C_n是G的回路，其中任何两个回路没有公共边。于是

$$G=C_1\cup C_2\cup\cdots\cup C_n$$

论点 3 得证。

$3\Longrightarrow1$　设G为边不相交回路C_1，C_2，…，C_n的并。考虑这些回路中的任何一个，比如说C_1。因为G是连通的，所以至少有一个回路（如C_2）与C_1有一个公共顶点v_1。让T_{12}为始于v_1的闭合轨迹，并连续经过C_1和C_2。这条轨迹明显地包含了C_1和C_2的所有边。

再者，因为G是连通的，所以T_{12}至少与一个回路（如C_3）有公共顶点v_2，而不同于C_1和C_2。开始于v_2的闭合轨迹T_{123}，连续经过T_{12}和C_3，将包含C_1、C_2和C_3的所有边。

重复这个过程，直至得到包含C_1、C_2、…、C_n所有边的闭合轨迹$T_{123\cdots n}$为止。这个闭合轨迹是G中一条欧拉轨迹。因此G是欧拉图。　□

根据这个定理，图3.3(a)的图G_1是欧拉图，因为G_1中每个顶点是度是偶数。而图3.3(b)和(c)的图G_2和G_3不是欧拉图，因为它们包含有奇度顶点（顶点的度是奇数）。也可以证明，欧拉图G_1是边不相交回路的并，这些回路的边集由下面给出：

$$\{e_4,\ e_5,\ e_6\}$$
$$\{e_3,\ e_7,\ e_8\}$$
$$\{e_2,\ e_9,\ e_{10}\}$$
$$\{e_1,\ e_{11},\ e_{12}\}$$

下面结果是定理3.1中论点 3 的推论。

推论3.1.1　欧拉图的每个顶点被包含在某个回路中。　□

虽然欧拉轨迹不存在于包含有奇度顶点的图中，但在这样的图中，可能构造一个边不相交的开轨迹集合，它们合在一起包含了图的所有边。这由下面定理证明。

定理3.2　设$G=(V,E)$为一个有$2k$个奇度顶点的连通图，$k\geqslant1$。那么E就能够划分成子集E_1，E_2，…，E_k，并使得每个E_i构成一条G的开轨迹。

证明

令r_i和s_i为G的$2k$个奇度顶点，$1\leqslant i\leqslant k$。现在对G加上k个新顶点w_1，w_2，…，w_k以及$2k$个边（r_i，w_i）和（s_i，w_i），$1\leqslant i\leqslant k$。在所产生的图G'中，每个顶点的度是偶数，因此G是欧拉图。

在G'的任何欧拉轨迹中可以注意到，边（r_i，w_i）和（s_i，w_i）对任何$1\leqslant i\leqslant k$相继出现。移走这$2k$条边，将导致k个边不相交的G的开轨迹，并使G的每一条边正好在其中某一开轨迹中出现。这些开轨迹给出了所需的E的划分。　□

推论3.2.1　设G是仅有两个奇度顶点的连通图，那么G有一个开轨迹（它始于一个奇度顶点，终于另一个奇度顶点）包含了G的所有边。　□

例如，图3.3(b)的图G_2仅有两个奇度顶点v_6和v_3，而开轨迹$\{e_1$，e_2，e_3，e_4，e_5，

e_6, e_7, e_8, e_9, e_{10}, e_{11}, e_{12}, e_{13}}却包含了G_2的所有边。这条轨迹始于v_6，终于v_1。

图3.3(c)的图G_3有4个奇度顶点。这个图有两个边不相交的开轨迹，这两个开轨迹由下列边集构成：

{e_1, e_2, e_3, e_4, e_5}

{e_6, e_7, e_8, e_9, e_{10}, e_{11}, e_{12}}

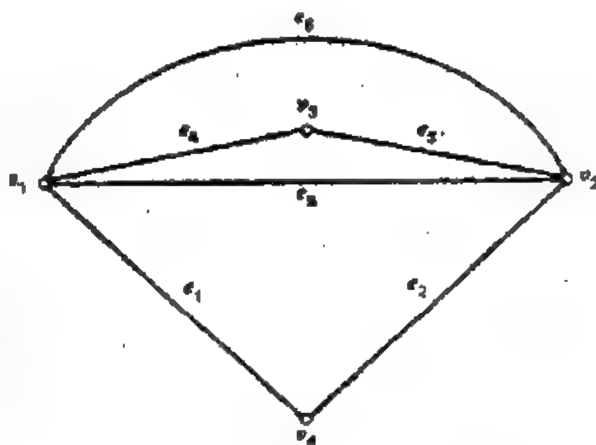


图 3.4 从两个顶点起的随机欧拉图

一个图G被说成是由顶点v起随机欧拉的，假如当我们从v出发任意沿着G的边走，最后总会得到一条欧拉轨迹。

应当注意，如果图G是由顶点v起随机欧拉的，那么把每个不包含全部边的闭合$v—v$轨迹扩展为G的欧拉轨迹是可能的。换言之，如果一个欧拉图G不是由顶点v起随机欧拉的，那么一定有一条包含所有与v关联的边、但不包含G的全部边的闭合$v—v$轨迹。

例如，考虑图3.4的欧拉图。这个图是由顶点v_1和v_2起随机欧拉的，从其他顶点起就不是随机欧拉的。可以证明，对于每个不同于v_1和v_2的顶点v_i，总存在一个包含所有与v_i关联的边、但不包括G的全部边的闭合$v_i—v_i$轨迹。例如，由边e_4，e_1，e_2和e_3组成的闭合$v_3—v_3$轨迹就具有这个性质。

下面定理给出了图的一个特性，此图是由顶点v起随机欧拉的。

定理3.3 一个欧拉图G是由顶点v起随机欧拉的，当且仅当G的每个回路都包含v。

证明

必要性 假设图G是由顶点v起随机欧拉的。若G中存在一个不包含v的回路C。考虑图$G'=G-C$，G'中每个顶点的度是偶数。G'可能是不连通的。但G''（G'中含v的连通片）是欧拉图，它包含了所有与v关联的边。这样，G''中存在一条始于和终于顶点v的欧拉轨迹T。这条轨迹必然包含所有与v关联的边，因此，它扩展到包括C的所有边是不可能的，这与G是由v起随机欧拉的假设相矛盾。

充分性 设欧拉图G中顶点v出现在G的每个回路中。假定G从v起不是随机欧拉的，那么G中存在一条包含与v关联的所有边，但不包含G的所有边的闭合$v—v$轨迹T。此外，还存在一个顶点$u \neq v$不是T中边的端点。

若从G中移走T的边，则产生了v是孤立顶点的图G'。在G'中，每个顶点的度是偶数，因此包含u的G'的连通片是一个欧拉图。根据推论3.1.1，G中有一个包含u的回路。这个回路显然不包含顶点v，这与v在G的每个回路中的假设相矛盾。 □

从图3.4的图G中可以证明，顶点v_1和v_2出现在G的每个回路中。这样，G是由这两个顶点起随机欧拉的。另一方面，对于其他的每个顶点总存在一个不包含本身的回路。

如果从图的每个顶点起都是随机欧拉的，这个图就是随机欧拉的。于是，由定理3.3可知，随机欧拉图G的所有顶点恰好都在G的一个回路C中，而在G中没有其他回路。换

言之，G是随机欧拉的，当且仅当它是一个回路。

3.2 哈密顿图

图G的一个哈密顿回路是一个包含G的全部顶点的回路。G的一条哈密顿路径是一条包含G的全部顶点的路径。

当图G含有一个哈密顿回路时被定义为哈密顿的。

图3.5（a）示出图G_1是哈密顿的，因为边序列e_1，e_2，e_3，e_4，e_5，e_6构成了G_1中的一个哈密顿回路。图3.5（b）有一条由边e_1，e_2，e_3，e_4形成的哈密顿路径，但没有哈密顿回路。

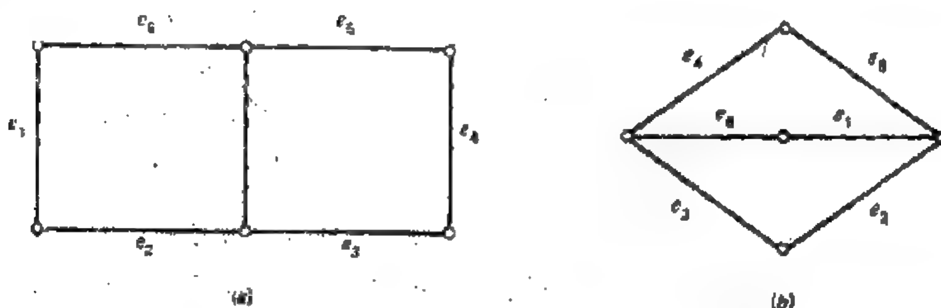


图 3.5 （a）一个哈密顿图；（b）有一条哈密顿路径的非哈密顿图。

一条欧拉轨迹是一条经过每条边恰好一次的闭合通道，而一条哈密顿回路是一条经过每个顶点恰好一次的闭合通道。这样，在一个欧拉图与一个哈密顿图之间有明显地相同之处。这可能使人们希望像欧拉图那样，存在一个简单、实用而精致的哈密顿图的特性。而事实并不如此，这种特性的研究在图论中是一个重要的悬而未决的问题。但是对一个哈密顿的简单图来讲，若干充分条件已经建立。（如果一个图既没有平行边，也没有自环，就叫做简单图）。本节将讨论这些条件的一部分。

一个序列$d_1 \leqslant d_2 \leqslant \cdots \leqslant d_n$被称为是图示的，如果在$n$个顶点$v_1$，$v_2$，…，$v_n$的图$G$中，对于每个$i$来说，顶点$v_i$的度$d(v_i)$等于$d_i$。（$d_1$，$d_2$，…，$d_n$）则称为$G$的度序列。

如果

$$S:\ d_1 \leqslant d_2 \leqslant \cdots \leqslant d_n$$

和

$$S^*:\ d_1^* \leqslant d_2^* \leqslant \cdots \leqslant d_n^*$$

是图示的序列，对于$1 \leqslant i \leqslant n$来讲，使得$d_i^* \geqslant d_i$那么$S^*$就说成是优化$S$。

下面的结果来自chvátal的文献[3.1]。

定理3.4 一个n阶简单图$G(V, E)$，其度序列为$d_1 \leqslant d_2 \leqslant \cdots \leqslant d_n$，若

$$d_k \leqslant k < \frac{1}{2}n \Longrightarrow d_{n-k} \geqslant n-k \qquad (3.1)$$

则图G是哈密顿的。

证明

首先注意，如果$d_k \leqslant k$，那么度不超过k的顶点数至少有k。类似地，如果$d_{n-k} \geqslant n-k$，那么度不超过$n-k$的顶点数至少是$k+1$。此外，如果一个图序列满足式（3.1），那么每个使其优化的图序列也满足式（3.1）。

现在用自相矛盾的方法来证明这个定理。

设有一个简单的非哈密顿图，它的度序列满足式（3.1），那么这个图是一个简单的最大非哈密顿图$G=(V, E)$的生成子图。图G的度序列$d_1 \leqslant d_2 \leqslant \cdots \leqslant d_n$也满足式（3.1）。

令u和v是G中两个不邻接的顶点，并使得$d(u)+d(v)$尽可能大，以及$d(u) \leqslant d(v)$。因为G是最大非哈密顿图，由此可知，加上一条连接u和v的边，将产生一个哈密顿图。这样，在G中有一条以u和v为端点的哈密顿路径$u-u_1$，u_2，$u_3 \cdots$，$u_n=v$（图3.6）。令

$$S=\{i \mid (u_1, u_{i+1}) \in E\}$$
$$T=\{i \mid (u_i, u_n) \in E\}$$



图 3.6

现在不存在$j \in S \cap T$。如果$j \in S \cap T$，那么边(u_1, u_{j+1})和(u_j, u_n)便在G中，循环顶点序列u_j，u_{j-1}，$\cdots$，u_1，u_{j+1}，u_{j+2}，$\cdots$，u_n，u_j将形成G中的一条哈密顿回路。

因为顶点$u_n=v$既不在S中也不在T中，所以$S \cup T \subseteq \{1, 2, \cdots, n-1\}$。

因此
$$d(u)+d(v)=|S|+|T|<n$$
和
$$d(u)<\frac{1}{2}n$$
这里$|X|$表示集X中的元素数。

因为$S \cap T=\varnothing$，$u_j (j \in S)$不与v邻接，于是$d(u)$和$d(v)$的选择，对$j \in S$暗指为$d(u_j) \leqslant d(u)$，这样，至少有$|S|=d(u)$个顶点，它们的度不超过$d(u)$。如若我们令$k=d(u)$。那么我们得到$d_k \leqslant k<\frac{1}{2}n$，按根据式（3.1）得$d_{n-k} \geqslant n-k$。这意味着，至少有$k+1$个顶点，它们的度不超过$n=k$。因为$d(u)=k$，那么顶点$u$最多能与这$k+1$个顶点中的$k$个顶点相邻。这样，就存在一个顶点$w$，它使$d(w) \geqslant n-k$，且不与$u$相邻。但这将使$d(u)+d(w) \geqslant n>d(u)+d(v)$与$d(u)$和$d(v)$的选择相矛盾。 □

在下面推论中，我们建立了一个图是哈密顿的充分条件。这些条件是由文献Dirac[3.2]，Ore[3.3]，Posa[3.4]与Bóndy[3.5]发展起来的。

推论3.4.1 一个简单图$G=(V, E)$，其$n \geqslant 3$，它的度序列为$d_1 \leqslant d_2 \leqslant \cdots \leqslant d_n$，如果满足下列条件之一，它就是哈密顿的：

1.(Driac)　$1 \leqslant k \leqslant n \Longrightarrow d_k \geqslant \frac{1}{2}n$

2.(Ore)　$(u, v) \notin E \Longrightarrow d(u)+d(v) \geqslant n$

3.(Pósa)　$1 \leqslant k < \frac{1}{2}n \Longrightarrow d_k > k$

4.(Bondy)　$j<k,\ d_j \leqslant j,\ d_k \leqslant k-1 \Longrightarrow d_j+d_k \geqslant n$

证明

我们将证明所有这些条件都意指式（3.1）。

1 $\Longrightarrow$ 2　任何满足条件 1 的度序列，显然也满足条件 2。

2 $\Longrightarrow$ 3　如果上述条件不成立，那么就存在一个 t，使得$1 \leqslant t < \frac{1}{2}n$和$d_t \leqslant t$。

现在假定存在一个$l(l<t)$及$(v_l, v_t) \notin E$，那么

$$d_l + d_t \leqslant 2d_t < 2 \cdot \frac{1}{2}n = n$$

与条件 2 相矛盾。因此，在顶点$v_1, v_2, \cdots, v_t$上导出的G的子图是一个完全图。

因为 $d_t \leqslant t$，每一个顶点 $v_i(1 \leqslant i \leqslant t)$，至多与一个顶点 $v_j(t+1 \leqslant j \leqslant n)$ 相邻。此外，$t < \frac{1}{2}n$意指$n-t>t$。这样存在一个顶点v_j，$t+1 \leqslant j \leqslant n$，它与任何$v_i(1 \leqslant i \leqslant t)$不相邻。这样，$d_j \leqslant n-t-1$。但另一方面

$$d_t + d_j \leqslant t+n-t-1$$

$$< n$$

这样我们有$(v_t, v_j) \notin E$及$d_t + d_j < n$，与条件 2 相矛盾。

3 $\Longrightarrow$ 4　如果这不成立，那么存在一个$j<k$，使得$d_j \leqslant j$，$d_k \leqslant k-1$及$d_j + d_k < n$。于是$d_j < \frac{1}{2}n$。如果我们现在建立$t = d_j < \frac{1}{2}n$，我们得到

$$d_t = d_{d_j} \leqslant d_j = t$$

因此我们有 $t < \frac{1}{2}n$和$d_t \leqslant t$，这与条件 3 相矛盾。

4 $\Longrightarrow$式（3.1）　如果这不成立，那么存在一个 t，使得 $d_t \leqslant t < \frac{1}{2}n$ 及 $d_{n-t} \leqslant n-t-1$。但另一方面

$$d_t + d_{n-t} \leqslant t+n-t-1$$

$$= n-1$$

与条件 4 相矛盾。　□

容易看出，如果一个图序列满足定理3.4和推论3.4.1中所列的任何一个条件，那么每个使其优化的图序列也是如此。这 5 个条件中最有力的是 chvátal 条件，它有个有趣的性质，即它是这类条件中最好的。如果一个图序列不能满足 chávtal 条件，那么它被一个非哈密顿图的度序列优化[3.1]。

虽然一般说来，建立一个图的非哈密顿性是困难的，但在某些情况下，利用一些巧妙的理由，这样做或许是可能的，下面通过例子（见文献[3.6]）来说明。

考虑图3.7所示的图G。我们证明：G中没有哈密顿路径。

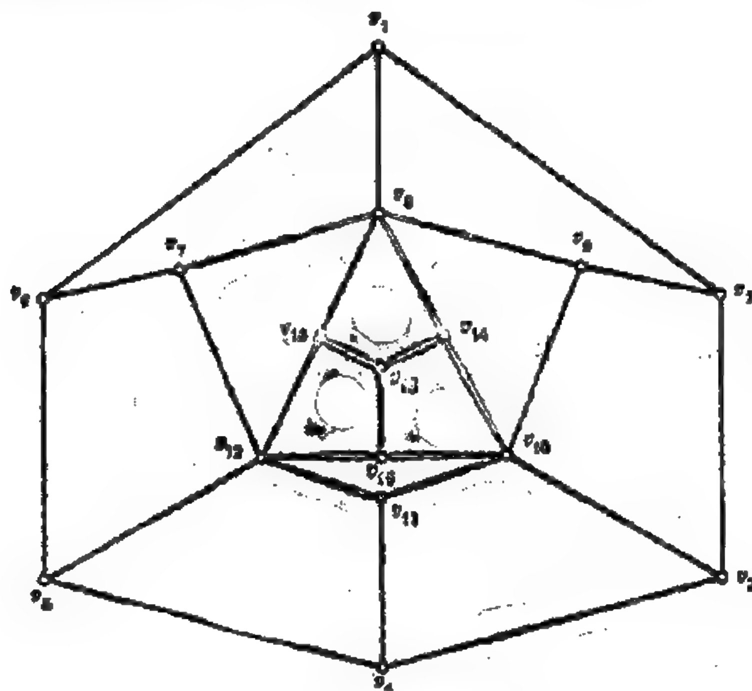


图 3.7 一个非哈密顿图

在与任何顶点相关联的所有边中，最多有两条边能包括在任何哈密顿路径中，在图G中，顶点v_8的度为5，因此与v_8相关联的边中，至少有三条不能包括在任何哈密顿路径中。顶点v_{10}和v_{12}也同样如此。因为v_2，v_4，v_6和v_{16}的度为3，则与其中任何一个顶点相关联的三条边中，至少有一条不包括在哈密顿路径中。这样，G中27条边至少有13条边不包括在任何哈密顿路径中。因此，在G的16个顶点上，没有足够的边来形成一条哈密顿路径。这样G没有哈密顿路径。

如果一个图从v出发的任何路径能够被扩展成一个哈密顿v—v回路，那么这个图就是从顶点v起随机哈密顿的。如果一个图从它的每个顶点起都是随机哈密顿的，那么这个图就是随机哈密顿的。

下面的定理完全描述了随机哈密顿图，这个定理的证明可参考文献[3.7]。

定理3.5 n阶简单图是随机哈密顿的，当且仅当它是一个回路或是一个完全图或是一个完全二分图$K_{n/2,n/2}$，最后一个只有当n是偶数时才为可能。 □

我们在结束本节时，来谈谈旅行商问题。该问题如下：

一个商人需要访问一些城市。如果他必须从家乡城市出发，访问每个城市恰好一次，然后返回家，他应该选取什么路线，使旅行距离最短？

假设我们用图的顶点代表城市，用连接顶点的边表示道路，路的长度可以表示为对应边的权。如果在任一对顶点之间有一条把他们连结起来的路，那么旅行商问题可以很明显

地等价于：在一个完全图上寻找一个最短的哈密顿回路，在该图上每条边都伴有一个权*。

在一个n阶完全图里，存在$(n-1)!/2$条哈密顿回路。一个求解旅行商问题的"蛮干"方法就是找出全部$(n-1)!/2$条哈密顿回路，然后挑出最短的。即使像对50这样小的n值，这种方法的工作量也是太大了（甚至对计算机来说也是如此）。对于任意的n值，没有现存的有效算法来求解这个问题。更深入的讨论，可参阅文献[3.8]、[3.9]、[3.10]和[3.11]。

3.3 进一步阅读

Berge文献[3.12]用一种统一方法证明了一个图是哈密顿的几个充分条件。该书也讨论了划分图的边集为路径、回路等问题。另外也可参看文献Bondy和Murty[3.13]。

3.2节中我们谈到，如果一个图的度序列不满足chvátal条件式（3.1），那么它就被一个非哈密顿图的度序列优化。此外，有些不满足式（3.1）的图序列，但它们必然是哈密顿图的度序列。参阅文献[3.14]文献[3.15]中推广了定理3.4。

有关哈密顿回路研究成果的综述，可参考文献[3.16][3.17]。

3.4 习　题

3.1 设G是一个有$2k$个奇度顶点的连通图。证明：G的边集E不能被划分成子集E_1，E_2，…，E_i，$i<k$，其中每个E_i形成G中的一条开欧拉轨迹。

3.2 证明：如果图G是从顶点v起随机欧拉的，那么v是唯一的割点或G没有割点。

3.3 设G是一个欧拉图，且有$n\geqslant 3$个顶点。证明：G是从一个顶点、两个顶点、全部顶点起随机欧拉的或G从任何顶点起都不是随机欧拉的。

3.4 如果G是从顶点v起随机欧拉的，那么证明：$\Delta(G)=d(v)$，这里$\Delta(G)$是G中最大度。

3.5 设G为从顶点v起的随机欧拉图。如果$d(u)=d(v)$，$u\neq v$，那么证明：G是从u起随机欧拉的。

3.6 是否存在含有一条欧拉轨迹，同时也是一个哈密顿回路的图？这种图有何特点。

3.7 证明：图3.8所示图没有哈密顿路径。

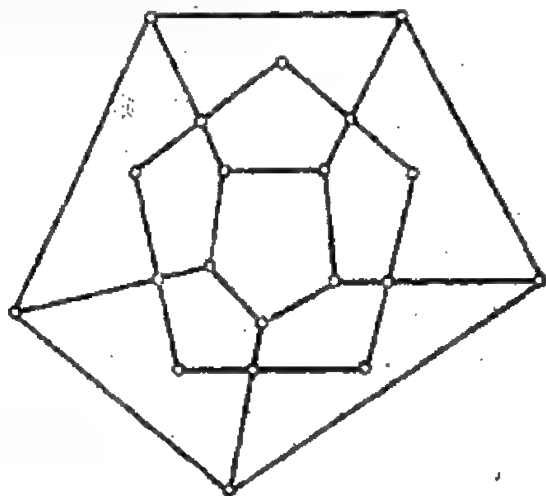


图 3.8

* 回路的权是回路中各边权的和。

3.8 设G是具有$n>2\delta(G)$个顶点的连通简单图。证明：G的最长路径应该大于或等于$2\delta(G)$。这里$\delta(G)$是G中的最小度（文献[3.2]）。

3.9 设G是一个有n个顶点的简单图。证明：如果对于G中每对顶点u和v来讲，$d(u)+d(v)\geq n-1$，那么G就有一条哈密顿路径。

3.10 令$d_1\leq d_2\leq\cdots\leq d_n$为一个有$n\geq 2$个顶点简单图$G$的度。如果

$$d_k\leq k-1\leq \frac{n}{2}-1\Rightarrow d_{n+1-k}\geq n-k$$

试证：G含有一个哈密顿路（Chaváta[3.1]）

3.11 设G是一个有n个顶点、m条边的简单图，且$n\geq 3$及$m\geq(n^2-3n+6)/2$，试证：G是哈密顿的（Ore[3.3]）。

3.12 设G是一个有n个顶点的简单图，u和v是G中两个不相邻的顶点，且

$$d(u)+d(v)\geq n$$

用G'表示加边（u，v）于G所得到的图。证明：G是哈密顿的，当且仅当G'是哈密顿的。

3.13 具有n个顶点的简单图G的闭包是这样的一个图，此图是由循环地连接G中不相邻的顶点对而得到的。每对顶点的度数和至少是n，直到没有剩余这样的顶点对为止。试证：图的闭包被完全确定。

注意　习题3.12的一个简单推论是，一个图是哈密顿的，当且仅当它的闭包是哈密顿的。若它的闭包是完全的，则G是哈密顿的这个结果可用来证明G是哈密顿的几个充分条件。参阅文献[3.15]和[3.13]。

3.14 证明：在完全图K_n中，一对边不相交的哈密顿回路的最大数目是$[(n-1)/2]$。

注意　$[x]$意指小于或等于x的最大整数。

3.15 如果在图G的每对相异顶点u和v间存在一条哈密顿$u-v$路径，则G是哈密顿-连通的。试证：若对G的每对不相邻顶点u和v，有$d(u)+d(v)\geq n+1$，则阶数$n\geq 3$的简单图是哈密顿-连通的（文献[3.18]）。

3.16 证明：连通图的树图（在习题2.28中定义）是哈密顿的（文献[3.19]和[3.20]）。

3.5 参考文献

3.1 V. Chvátal, "On Hamilton's Ideals," *J. Combinatorial Theory B*, Vol. 12, 163—168(1972).

3.2 G. A. Dirac, "Some Theorems on Abstract Graphs," *Proc. London Math. Soc.*, Vol. 2, 69—81(1952).

3.3 O. Ore, "Arc Coverings of Graphs," *Ann. Mat.Pura Appl.*, Vol. 55, 315—321 (1961).

3.4 L. Pósa, "A Theorem Concerning Hamilton Lines," *Magyar Tud. Akad. Mat Kutató Int. Közl.*, Vol. 7, 225—226(1962).

3.5 J. A. Bondy, "Properties of Graphs with Constraints on Degrees," *Studia Sci. Math. Hungar.*, Vol. 4, 473—475(1969).

3.6 C. L. Liu, *Introduction to Combinatorial Mathematics*, McGraw-Hill, New York, 1968.

3.7 M. Behzad and G. Chartrand, *Introduction to the Theory of Graphs*, Allyn and Bacon, Boston, 1971.

3.8 S. Lin, "Computer Solutions of the Traveling Salesman Problem," *Bell Syst. Tech J.*, Vol. 44, 2245—2269(1965)

3.9 M. Belmore and G. L. Nemhauser, "The Traveling Salesman Problem: A Survey," *Operations Res*, Vol. 16, 538—558(1968).

3.10 M. Held and R. M. Karp, "The Traveling Salesman Problem and Minimum Spanning Trees," *Operations Res.*, Vol. 18, 1138—1162(1970).

3.11 M. Held and R.M. Karp, "The Traveling Salesman Problem and Minimum Spanning Trees: Part II," *Math. Programming*, Vol. 1, 6—25(1971).

3.12 C. Berge, *Graphs and Hypergraphs*, North Holland, Amsterdam, 1973.

3.13 J.A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, Macmillan, London, 1976, chap.4.

3.14 C. St. J. A. Nash-Williams, "Hamilton Arcs and Circuits," in *Recent Trends in Graph Theory*, Springer, Berlin, 1971, pp. 197—210.

3.15 J. A. Bondy and V. Chvátal, "A Method in Graph Theory," *Discrete Math.*, Vol. 15, 111—135(1976).

3.16 C. St. J.A. Nash-Williams, "Hamiltonian Circuits," in *Studies in Graph Theory*, Part II, MAA Press, 1975, pp. 301—360.

3.17 L. Lesniak-Foster, "Some Recent Results in Hamiltonian Graphs," *J. Graph Theory*, Vol. 1, 27—36(1977).

3.18 P. Erdös and T. Gallai, "On Maximal Paths and Circuits of Graphs," *Acta Math. Acad. Sci. Hung.*, Vol. 10, 337—356(1959).

3.19 R.L. Cummins, "Hamilton Circuits in Tree Graphs," *IEEE Trans. Circuit Theory*, Vol. CT-13, 82—60(1966)

3.20 H. Shank, "A Note on Hamilton Circuits in Tree Graphs," *IEEE Trans. Circuit Theory*, Vol. CT-15, 86(1968).

第四章　图和矢量空间

近世代数是科学家和工程师研究问题的极为有效工具。因为属于这种代数结构的一些有力的精致的理论可以用来研究事物的集合，所以判明这些事物集合的代数结构就认为是很必要了。系统理论、电网络理论、编码理论、开关电路（时序的和组合的）以及计算机科学等就是几个应用这种方法并大受其益的领域。

本章将证明，矢量空间能够与图相关联，并详细讨论矢量空间的两个重要子空间的性质，即割集空间和回路空间的性质。在本书的第二篇将会看到，这些子空间确实定义了电网络的电压和电流空间。

本章的前两节，介绍了后面几节要用到的代数的基本概念和结论。进一步详细讨论线性代数的这些概念和结论，可参阅文献[4.1]、[4.2]和[4.3]。

4.1　群　和　域

考虑一个有限集$S=\{a, b, c, \cdots\}$。令＋表示定义在S上的二元运算。这个运算对S中的每一对元素a和b给定了一个唯一元素$a+b$。如果a和b在S中，且元素$a+b$也属于S，则称S在＋下是闭合的。

如果，对于S中所有的a、b和c，

$$a+(b+c)=(a+b)+c,$$

则称运算＋是结合的。

进而如果对于S中所有的a和b，

$$a+b=b+a,$$

则称运算＋是可交换的。

现在我们来定义群。

如果下面的假设成立：

1. S在＋下是闭合的。

2. ＋是结合的。

3. 若S中存在唯一元素e，对于S中所有的a，使得$a+e=e+a=a$，则元素e叫做群的幺元。

4. 对于S中任一元素a，若存在唯一元素b，使得$b+a=a+b=e$，则元素b叫做a的逆元，反之亦然。显然，幺元e是它自身的逆元。

则称一个带有二元运算＋（叫做加法）的集S是一个群。

如果运算＋是可交换的，这个群就是阿贝尔群。

群的最常见的例子是带有＋（定义为普通加法运算）的全体整数集合$S=\{\cdots, -2, -1, 0, 1, 2, \cdots\}$。在这个群中，0是幺元，对于$S$中所有的$a$，$-a$是$a$的逆元。注

意，这个群是阿贝尔群。

带有乘法运算的全体整数集合 S 是一个群吗？（不是。为什么？）

群的一个重要例子是带有模*P 加法运算的整数集合 $Z_p=\{0, 1, 2, \cdots, p-1\}$。在这个群中，0 是幺元，并且对于所有不等于 0 的a，整数$p-a$是整数 a 的逆元。当然，0 是它自身的逆元。这个群也是阿贝尔群。举例，给出了Z_5的加法表：

+	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

下面来定义域：

如果下列假设满足：

1. F 是 + 下的阿贝尔群，其中幺元记为 e。
2. 集合$F-\{e\}$是乘法运算下的阿贝尔群。
3. 乘法运算对加法运算是可分配的，即

$a\cdot(b+c)=(a\cdot b)+(a\cdot c)$，对于 F 中所有的 a，b 和 c。

则称带有两种运算 + 和 · （分别叫加法和乘法）的集合 F 是域。举例，再来考虑带有加法（模 P）和乘法（模 P）两种运算的集合$Z_p=\{0, 1, 2, \cdots, p-1\}$。如前所示，$Z_p$是模 P 加法下的阿贝尔群，以 0 为其幺元。可以证明当且仅当 P 是素数时，在模 P 乘法下，集合$Z_p-\{0\}=\{1, 2, \cdots, p-1\}$是群，也就是阿贝尔群。模 P 乘法对模 P 加法是可分配的这一事实是容易证明的。这样，当且仅当 P 是素数时，集合Z_p是域。

域Z_p通常记为$GF(p)$，并称为Galois域。作为例子，下面给出 $GF(5)$的乘法表：

·	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

我们特别感兴趣的域是模 2 整数的集合$GF(2)$。在这个域中，

$0+0=0 \qquad 0\cdot 0=0$

$1+0=0+1=1 \qquad 1\cdot 0=0\cdot 1=0$

$1+1=0 \qquad 1\cdot 1=1$

* 如果$a=mp+q$，$0\leq q\leq p-1$，则在模计算中$a=q$(模p)。

4.2 矢量空间

考虑带有二元运算⊞的集合S。令F是带有＋和·的域，其中＋和·分别表示加法和乘法运算。在F中的元素和S中的元素之间也定义了乘法运算，用$*$表示。这个运算对每一有序对（α, s）给定了S中的一个唯一元素，记为$\alpha * s$，其中α属于F，s属于S。如果下面假设成立：

1. S是⊞下的阿贝尔群。

2. 对于F中的任何元素α和β，以及S中的任何元素s_1和s_2，

$$\alpha*(s_1 \boxplus s_2)=(\alpha*s_1)\boxplus(\alpha*s_2)$$

和

$$(\alpha+\beta)*s_1=(\alpha*s_1)\boxplus(\beta*s_1)。$$

3. 对于F中的任何元素α和β，以及S中的任何元素s，

$$(\alpha\cdot\beta)*s=\alpha*(\beta*s)$$

4. 对S中的任何元素s，$1*s=s$，其中1是F中的乘幺元。

则集合S叫做在F上的一个矢量空间。

下面给出矢量空间的一个重要例子。

考虑域F上所有n-矢量*的集合W（注意，n-矢量的元素取自F）。符号＋和·分别代表F中的加法和乘法运算，符号0和1分别表示F中的加幺元和乘幺元。令⊞是W上的加法运算，$*$是W中的元素和F中元素之间的乘法运算，其定义如下：

1. 如果$w_1=(\alpha_1, \alpha_2, \cdots, \alpha_n)$和$w_2=(\beta_1, \beta_2, \cdots \beta_n)$是$W$中的元素，那么

$$w_1 \boxplus w_2=(\alpha_1+\beta_1, \alpha_2+\beta_2, \cdots, \alpha_n+\beta_n)$$

2. 如果α属于F，那么

$$\alpha*w_1=(\alpha\cdot\alpha_1, \alpha\cdot\alpha_2, \cdots, \alpha\cdot\alpha_n)$$

根据上面定义的⊞，可以推出W是⊞下的阿贝尔群，其中以n-矢量（0, 0, 0, …, ）为幺元素。这样，W满足矢量空间定义的第一个假设。我们也容易证明，W和F中的元素满足矢量空间的其它三个要求。

例如，下面给出的八个3-矢量集合W是$GF(2)$上的矢量空间：

$$w_0=(000),\ w_1=(001),\ w_2=(010),\ w_3=(011)$$

$$w_4=(100),\ w_5=(101),\ w_6=(110),\ w_7=(111)$$

这个矢量空间在本节所有的例子中都要用到。

下面叙述与矢量空间有关的几个重要定义和结论（不加以证明）。

考虑域F上的矢量空间S。

矢量和标量　S中的元素称为矢量，F中的元素称为标量

线性组合　如果S中的元素s可表示为

$$s=(\alpha_1*s_1)\boxplus(\alpha_2*s_2)\boxplus\cdots\boxplus(\alpha_j*s_j)$$

---

* F上的n-向量是含有F中n个元素的行矢量。

其中s_i是矢量，a_i是标量，那么s就叫做$s_1, s_2, \cdots, s_j$的线性组合。

线性独立 矢量$s_1, s_2, \cdots, s_j$是线性独立的，如果这个集中没有一个矢量可表示为其它矢量的线性组合。

基矢量 矢量$s_1, s_2, \cdots, s_n$构成矢量空间S的一个基，如果它们是线性独立的，且S中的每一个矢量都可表示成这些矢量的线性组合。矢量$s_1, s_2, \cdots, s_n$称为基矢量。

对于给定的基可以证明，一个由基矢量线性组合成的矢量表达式是唯一的。一个矢量空间可以有不止一个基。然而还可以证明所有的基都含有相同数目的矢量。

维数 矢量空间S的维数是S中一个基的矢量数，记为$\dim(S)$

子空间 如果S是F上矢量空间S的子集，且S'又是F上的矢量空间，那么S'就是S的子空间。

直接和 S的两个子空间S_1和S_2的直接和$S_1 \boxplus S_2$是形如$s_i \boxplus s_j$的所有矢量的集合，其中s_i属于S_1，s_j属于S_2。

可以证明，$S_1 \boxplus S_2$也是一个子空间，它的维数由

$$\dim(S_1 \boxplus S_2) = \dim(S_1) + \dim(S_2) - \dim(S_1 \cap S_2)$$

给出。注意，当S_1和S_2是子空间时，$S_1 \cap S_2$也是子空间。

我们用$GF(2)$上3-矢量空间W来解释上面的定义。W中的矢量$w_0, w_1, \cdots, w_7$定义同前。

1.因为

$$w_1 = (1 * w_6) \boxplus (1 * w_7)$$

所以，w_1是w_6和w_7的线性组合。

2.因为w_3和w_4不能用一个来表示另一个，所以矢量w_3和w_4是线性独立的。注意，对任意的i，w_0和w_i并不是线性独立的。

3.因为w_1、w_2和w_4是线性独立的，且其余的矢量都可表示为它们的线性组合：

$$w_0 = (0 * w_1) \boxplus (0 * w_2) \boxplus (0 * w_4)$$
$$w_3 = (1 * w_1) \boxplus (1 * w_2)$$
$$w_5 = (1 * w_1) \boxplus (1 * w_4)$$
$$w_6 = (1 * w_2) \boxplus (1 * w_4)$$
$$w_7 = (1 * w_1) \boxplus (1 * w_2) \boxplus (1 * w_4)$$

所以矢量w_1，w_2和w_4构成了W中的一个基。

可以证明，矢量w_1，w_3和w_7也构成了一个基。

4.因为W的基中有三个矢量，所以W的维数等于3。

5.集合

$$W' = \{w_0, w_1, w_2, w_3\}$$

和

$$W'' = \{w_0, w_1, w_6, w_7\}$$

是W的子空间。可以证明，$\{w_1, w_2\}$是W'的基，$\{w_1, w_6\}$是W''的基。于是

$$\dim(W') = 2$$

和

$$\dim(W'')=4$$

6.如果W'和W''的定义同上，那么

$$W'⊞W''=\{w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7\}$$

7.$W'⊞W''$的维数由下式给出：

$$\dim(W'⊞W'')=\dim(W')+\dim(W'')-\dim(W'\cap W'')$$
$$=4-\dim(W'\cap W'')$$

因为

$$W'\cap W''=\{w_0, w_1\}$$

我们有

$$\dim(W'\cap W'')=1$$

于是

$$\dim(W'⊞W'')=4-1=3$$

注意，在$W'⊞W''=W$的情况下，也能得到上式。

下面我们来定义在相同域上定义的两个矢量空间的同构。

令S和S'是域F上两个n维矢量空间。如果S和S'存在一一对应关系，并使下面假设成立：

1.如果S的矢量s_1和s_2分别对应于S'的矢量s_1'和s_2'，那么矢量$s_1⊞s_2$对应于矢量$s_1'⊞s_2'$，其中⊞和⊞是在S和S'中对应的运算。

2.对于F中任意α，如果s对应于s'，矢量$\alpha * s$对应于矢量α⚠s'，其中$*$和⚠是在s和s'中对应的运算。

则称S和S'是同构的。

考虑域F上n维矢量空间S和n-矢量的n维矢量空间W。令矢量s_1，s_2，…，s_n构成S中的一个基。假若我们定义S中的矢量s对应于W中矢量$w=(\alpha_1, \alpha_2, \cdots, \alpha_n)$*当且仅当$s=(\alpha_1 * s_1)⊞(\alpha_2 * s_2)⊞\cdots⊞(\alpha_n * s_n)$。那么不难证明，这种一一对应关系定义了$S$和$W$之间的同构。于是，有下面的重要结论。

定理4.1 域F上的每个n维矢量空间与F上n-矢量的矢量空间W同构。 □

上述定理提供了矢量空间和矩阵之间的主要联系。它的含义是域F上的n维矢量空间可以用F上所有n-矢量的n维矢量空间W来进行研究。

结束本节之前，再介绍两个重要概念的定义——点积（或内积）和正交性。

令

$$w_1=(\alpha_1, \alpha_2, \cdots, \alpha_n)$$

以及

$$w_2=(\beta_1, \beta_2, \cdots, \beta_n)$$

是F上n-矢量的矢量空间W中的两个矢量。w_1和w_2的点积是一个如下定义的标量，记为

---

* α_1，…，α_n叫作相对于基{s_1，s_2，…s_n}的坐标。

$<w_1, w_2>$

$$<w_1, w_2> = \alpha_1\cdot\beta_1+\alpha_2\cdot\beta_2+\cdots+\alpha_n\cdot\beta_n$$

举例，如果

$$w_1 = (0\ 1\ 0\ 0\ 1)$$

和

$$w_2 = (1\ 0\ 1\ 1\ 1)$$

那么

$$\begin{aligned}<w_1, w_2> &= 0\cdot1+1\cdot0+0\cdot1+0\cdot1+1\cdot1\\ &= 0+0+0+0+1\\ &= 1\end{aligned}$$

矢量w_i和w_j是相互正交的，如果$<w_i, w_j> = 0$，其中0是F的加么元。例如，矢量

$$w_1 = (1\ 1\ 0\ 1\ 1)$$

和

$$w_2 = (1\ 1\ 1\ 0\ 0)$$

在$GF(2)$上是正交的，因为

$$\begin{aligned}<w_1, w_2> &= 1\cdot1+1\cdot1+0\cdot1+1\cdot0+1\cdot0\\ &= 1+1+0+0+0\\ &= 0\end{aligned}$$

W的两个子空间W'和W''是W的正交子空间，如果一个子空间中的任一矢量与另一子空间中每个的矢量都正交。

W的两个子空间W'和W''是正交互补的，如果它们是相互正交，且它们的直接和$W'\oplus W''$等于矢量空间W。

例如，再来考虑$GF(2)$上3-矢量的3维矢量空间W。在这个矢量空间中，子空间$W' = \{w_0, w_1, w_2, w_3\}$和$W'' = \{w_0, w_4\}$是相互正交的。可以证明，$W'$和$W''$的直接和等于$W$。因此，$W'$和$W''$是正交互补的。

4.3 图的矢量空间

本节将说明怎样使一个矢量空间与一个图相关联，并指出这个矢量空间的两个重要子空间。

考虑图$G=(V, E)$。令W_G是E的所有子集的集合，其中包括空集$\varnothing$。首先证明W_G是⊕下的阿贝尔群⊕是集合间的环和运算。在适当定义了域$GF(2)$和W_G元素之间的乘法后，就可证明W_G是$GF(2)$上的矢量空间。

下面的论点是容易证明的：

1. W_G在⊕下是闭合的。

2. ⊕是结合的，

3. ⊕是可交换的。

而且，对W_G中任何元素E_i，

$$E_i \oplus \varnothing = E_i$$

和

$$E_i \oplus E_i = \varnothing$$

于是，对于运算⊕，ϕ是幺元，E_i是它自身的逆元。因此W_G是⊕下的阿贝尔群，并满足矢量空间定义中的第一条要求。

令 * 为$GF(2)$和W_G元素间的乘法运算，并且定义如下：

对于W_G中任意E_i，

$$1 * E_i = E_i$$

以及

$$0 * E_i = \varnothing$$

运用 * 的定义，就容易证明W_G的元素满足矢量空间的如下几个要求：

对于$GF(2)=\{0,1\}$中的任意元素α和β，以及W_G中的任意元素E_i和E_j，有

1. $(\alpha+\beta) * E_i = (\alpha * E_i) \oplus (\beta * E_i)$。
2. $\alpha * (E_i \oplus E_j) = (\alpha * E_i) \oplus (\alpha * E_j)$。
3. $(\alpha \cdot \beta) * E_i = \alpha * (\beta * E_i)$。
4. $1 * E_i = E_i$。

（注意，1是$GF(2)$中的乘幺元。）

所以W_G是$GF(2)$上的矢量空间。

如果$E=\{e_1, e_2, \cdots, e_m\}$，那么显而易见，子集$\{e_1\}$，$\{e_2\}$，…，$\{e_m\}$构成为$W_G$的一个基。于是$W_G$的维数是$m$，即$G$中的边数。

因为G的每个边导出子图对应于E的一个唯一子集，并且由定义（见第一章），任意两个边导出子图的环和对应于它们相应边集的环和。所以很显然，G的所有导出子图的集合也是$GF(2)$上的矢量空间，如果我们定义乘法运算 * 为：

对G的任何边导出子图G_i，

$$1 * G_i = G_i$$

及

$$0 * G_i = \varnothing$$

这是不包含顶点和边的空图。

这个矢量空间也可用符号W_G来表示。注意W_G包括了空图$\varnothing$。

下面的定理总结了上面讨论的结果。

定理4.2 对于含有m条边的图G，W_G是$GF(2)$上的m维矢量空间。 □

本章中，因为我们只涉及边导出子图，所以就把边导出子图简称为子图，而略去形容词“边导出”。但是，在有些地方仍然用这个形容词来强调有关子图的边导出性质。

下面证明W_G的下列子集是子空间：

1. W_C，即G中所有回路（包括空图ϕ）和边不相交回路的并的集合。
2. W_S，即G中所有割集（包括空图ϕ）和边不相交割集的并的集合

一旦我们证明了W_C和W_S在⊕（即环和运算）下是闭合的，就可推出结论。

定理4.3 W_C，即图G的所有回路和边不相交回路的并的集合，是G的矢量空间W_G的子空间。

证明

由定理3.1，一个图可以表示成边不相交回路的并，当且仅当图中每个顶点的度数为偶数。因此，可以把W_C当作G的所有边导出子图的集合，其中所有顶点的度数都为偶数。

考虑W_C中任意两个不同部分C_1和C_2，C_1和C_2是所有顶点的度数为偶数的边导出子图。令C_3表示C_1和C_2的环和。为了证明此定理，只需证明C_3属于W_C，即证明C_3中每个顶点的度数都是偶数。

考虑C_3中的任一顶点v。显然，这个顶点至少要在子图C_1和C_2的某一个中出现。令X_i表示C_i中与v相关联的边的集合，$i=1, 2, 3$。令$|X_i|$表示X_i中的边数。于是$|X_i|$是C_i中v的度数。由于$|X_1|$和$|X_2|$都是偶数，其中某一个可能为零，而$|X_3|$不为零。

因为$C_3=C_1\oplus C_2$，我们得到

$$X_3=X_1\oplus X_2$$

于是

$$|X_3|=|X_1|+|X_2|-2|X_1\cap X_2|$$

因为$|X_1|$和$|X_2|$都是偶数，从上述方程明显得出$|X_3|$是偶数，即C_3中v的度数为偶数。因为这对C_3中所有顶点均成立，从而得出C_3属于W_C，定理得证。 □

W_C叫做图G的回路子空间。

举例来解释上面定理的结论，考虑图4.1(a)所示图G。图4.1(b)和(c)示出的子图C_1和C_2是G的边不相交回路的并，因为图中每个顶点的度数均为偶数。这样，C_1和C_2均属于W_C的子空间W_C。C_1和C_2的环和C_3，如图4.1(d)所示，显而易见，C_3的所有顶点的度数均为偶数。因此，C_3也属于W_C。

下面来证明G的所有割集和边不相交割集的并的集合W_S，也是W_G的子空间。

由定理2.7可知，一个切割是一个割集，或者是一些边不相交割集的并。于是G的每个切割都属于W_S。现在来证明W_S的每个元素都是一个切割。证明了这一点，也就是证明了W_S是W_G的一个子空间。

定理4.4 图G中任意两个切割的环和仍是图G中的一个切割。

证明

考虑图$G=(V,E)$中任意两个切割$S_1=\langle V_1,V_2\rangle$，$S_2=\langle V_3,V_4\rangle$。注意

$$V_1\cup V_2=V_3\cup V_4=V$$

及

$$V_1\cap V_2=V_3\cap V_4=\varnothing$$

令

$$A=V_1\cap V_3$$
$$B=V_1\cap V_4$$
$$C=V_2\cap V_3$$

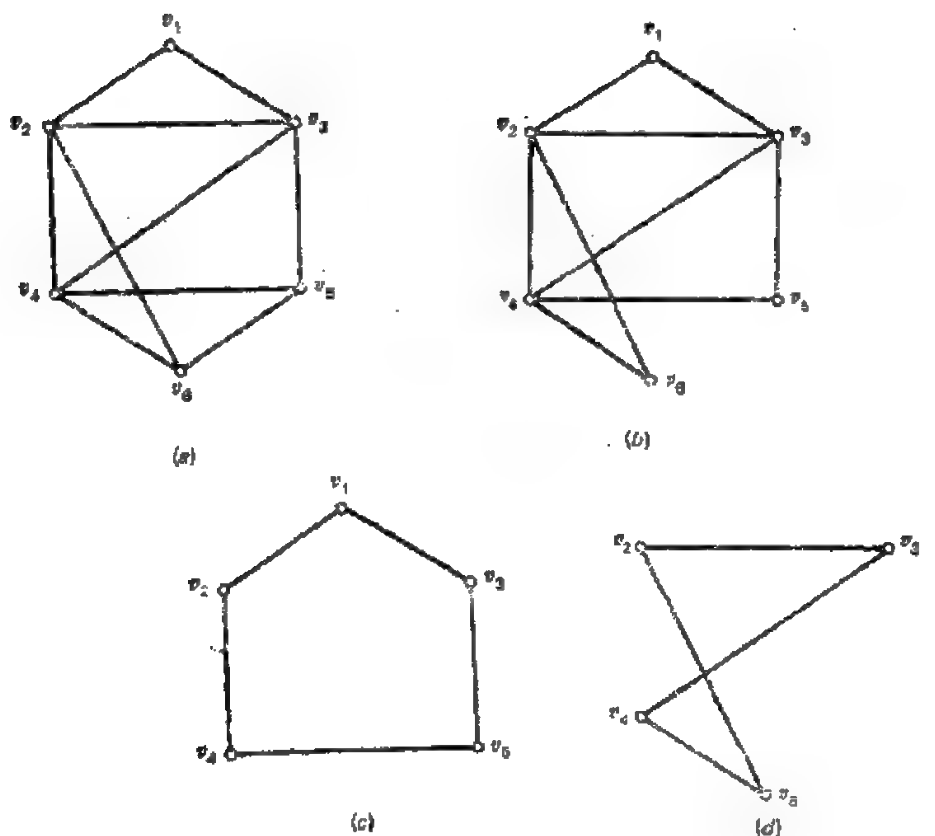


图 4.1 (a)图G; $(b)G$的子图C_1; $(c)G$的子图C_2; $(d)G$的子图$C_1\oplus C_2$。

$$D=V_2\cap V_4$$

很容易看出，集合A, B, C和D是互不相交的。因此

$$S_1=\langle A\cup B,\ C\cup D\rangle$$
$$=\langle A,\ C\rangle\cup\langle A,\ D\rangle\cup\langle B,\ C\rangle\cup\langle B,\ D\rangle$$

及

$$S_2=\langle A\cup C,\ B\cup D\rangle$$
$$=\langle A,\ B\rangle\cup\langle A,\ D\rangle\cup\langle C,\ B\rangle\cup\langle C,\ D\rangle$$

于是，得到

$$S_1\oplus S_2=\langle A,\ C\rangle\cup\langle B,\ D\rangle\cup\langle A,\ B\rangle\cup\langle C,\ D\rangle$$

因为

$$\langle A\cup D,\ B\cup C\rangle=\langle A,\ C\rangle\cup\langle B,\ D\rangle\cup\langle A,\ B\rangle\cup\langle C,\ D\rangle$$

所以$S_1\oplus S_2$可以写成

$$S_1\oplus S_2=\langle A\cup D,\ B\cup C\rangle$$

因为$A\cup D$和$B\cup C$是互不相交的，而且二者包含了V中所有的顶点，所以$S_1\oplus S_2$是G的一个切割。于是定理得证。 □

因为两个不相交集合的环和等于它们的并，于是得出上述定理的如下推论。

推论4.4.1 图G的任何两个边不相交切割的并仍是G中的一个切割。 □

因为割集也是切割，显然由推论4.4.1得出，W_S是G中所有切割的集合。

进而，由定理4.4可知，W_S在环和运算下是闭合的。于是，得到了下面的定理。

定理4.5 G中所有割集和边不相交割集的并的集合W_S，是G的矢量空间W_G的子空间。 □

W_S叫做图G的割集子空间。

举例来解释定理4.5的结论，考虑图4.2所示图G中的割集S_1和S_2：

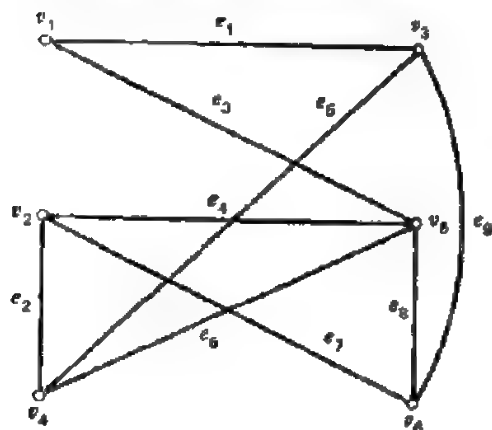


图 4.2

$$S_1=\{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}=\langle V_1, V_2\rangle$$

$$S_2=\{e_1, e_2, e_4, e_5, e_8\}=\langle V_3, V_4\rangle$$

其中

$$V_1=\{v_1, v_2, v_4\}$$
$$V_2=\{v_3, v_5, v_6\}$$
$$V_3=\{v_1, v_4, v_5\}$$
$$V_4=\{v_2, v_3, v_6\}$$

因此

$$S_1\oplus S_2=\{e_2, e_3, e_6, e_7, e_8\}$$

如果

$$A=V_1\cap V_3=\{v_1, v_4\}$$
$$B=V_1\cap V_4=\{v_2\}$$
$$C=V_2\cap V_3=\{v_5\}$$
$$D=V_2\cap V_4=\{v_3, v_6\}$$

那么可以证明，集合$S_1\oplus S_2=\{e_2, e_3, e_6, e_7, e_8\}$能够写成（如定理4.4所证明的那样）为

$$S_1\oplus S_2=\langle A\cup D, B\cup C\rangle=\langle\{v_1, v_4, v_3, v_6\}, \{v_2, v_5\}\rangle$$

4.4 回路和割集子空间的维数

本节将证明回路和割集子空间的维数分别等于图G的零度和秩。要证明这一点，只需证明对应于连通图某个生成树的基本回路集和基本割集集，分别为图的回路和割集子空间的基。

令T是具有n个顶点、m条边的连通图G的一个生成树。T的树支记为$b_1, b_2, \cdots, b_{n-1}$，连支记为$c_1, c_2, \cdots, c_{m-n+1}$。令$C_i$和$S_i$分别表示对应于$c_i$和$b_i$的基本回路和基本割集。

由定义，每个基本回路仅包含一条连支，这条连支又不出现在其它基本回路中。于是，没有一个基本回路可表示成其它基本回路的环和，因而，基本回路$C_1, C_2, \cdots, C_{m-n+1}$是独立的。同样，因为每个基本割集只包含一个树支，这个树支又不出现在其它基

本割集中。所以基本割集S_1，$S_2\cdots$，S_{n-1}也是独立的。

为了证明C_1，$C_2\cdots$，C_{m-n+1}（S_1，$S_2\cdots$，S_{n-1}）构成了G的回路（割集）子空间的一个基，我们只需证明G的回路(割集)子空间中的每个子图可表示成诸C_i（S_i）的环和。

考虑G的回路子空间的任意子图C。假定C包含连支c_{i_1}，c_{i_2}，c_{i_r}。令C'表示基本回路C_{i_1}，C_{i_2}，…，C_{i_r}的环和。显然，连支c_{i_1}，c_{i_2}，…，c_{i_r}出现在C'中，C'不再含有T中其它连支。因为C也包含这些连支，而不包含这些连支，所以$C'\oplus C$不含连支。

现在来证明$C'\oplus C$为零。如果它不成立，那么由前面的讨论，$C'\oplus C$仅包含树支，而无回路。另一方面，由于$C'\oplus C$是回路的环和，由定理4.3，它就是回路或边不相交回路的并。于是，这与$C'\oplus C$非零的假定产生矛盾。所以$C'\oplus C$是零。这就意味着$C=C'=C_{i_1}\oplus C_{i_2}\oplus\cdots\oplus C_{i_r}$。即G的回路子空间中的每个子图都可表示成诸$C_i$的环和。

用类似的方法可以证明，G的割集子空间中的每个子图都可表示成诸S_i的环和。

于是，有下面的定理。

定理4.6 设连通图G具有n个顶点，m条边。那么

1.对应于G的一个生成树的基本回路构成了G^*的回路子空间的一个基，而且G的回路子空间的维数等于$m-n+1$，即G的零度。

2.对应于G的一个生成树的基本割集构成了G的割集子空间的一个基，而且G的割集子空间的维数等于$n-1$，即G的秩。 □

显而易见，在图G是不连通的情况下，在对应于G的一个林中连支的所有基本回路集合和树支的所有基本割集集合，分别构成了G的回路和割集子空间的基。于是得到上述定理的如下推论：

推论4.6.1 如果图G具有n个顶点，m条边和p个片，那么

1.G的回路子空间的维数等于G的零度$m-n+p$。

2.G的割集子空间的维数等于G的秩$n-p$。 □

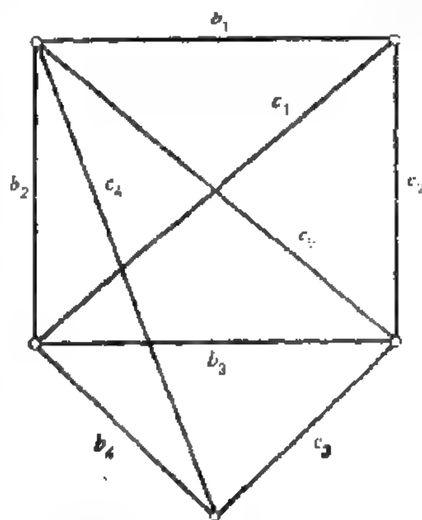


图 4.3

举例来考虑图4.3所示图G。由b_1，b_2，b_3，b_4标记的边构成了G的一个生成树。T的连支标记为c_1，c_2，c_3，c_4和c_5。所以对应于连支c_1，c_2，c_3，c_4和G_5的基本回路C_1，C_2，C_3，C_4和C_5以及对应于树支b_1，b_2，b_3，b_4的基本割集S_1，S_2，S_3和S_4得出如下：

$$C_1=\{c_1,\ b_1,\ b_2\}\qquad S_1=\{b_1,\ c_1,\ c_2\}$$
$$C_2=\{c_2,\ b_1,\ b_2,\ b_3\}\qquad S_2=\{b_2,\ c_1,\ c_2,\ c_4,\ c_5\}$$
$$C_3=\{c_3,\ b_3,\ b_4\}\qquad S_3=\{b_3,\ c_2,\ c_3,\ c_5\}$$
$$C_4=\{c_4,\ b_2,\ b_4\},\qquad S_4=\{b_4,\ c_3,\ c_4\}$$
$$C_5=\{c_5,\ b_2,\ b_3\}$$

首先考虑由边b_1，b_2，b_3，c_2，c_3，c_4和c_5组成的子图C。可以证明，C中每个顶点度数均为偶数，因此C属于G的回路子空间。因为C包含连支c_2，c_3，c_4和c_5，则由定理

4.6以前的讨论可得，C应等于基本回路C_2，C_3，C_4和C_5的环和，其正确性可证明如下：

$$\begin{aligned} C_2\oplus C_3+C_4\oplus C_5&=\{c_2,\ b_1,\ b_2,\ b_3\}\oplus\{c_3,\ b_3,\ b_4\}\oplus\{c_4,\ b_1,\ b_4\}\oplus \\ &\quad\{c_5,\ b_2,\ b_3\} \\ &=\{b_1,\ b_2,\ b_3,\ c_2,\ c_3,\ c_4,\ c_5\} \\ &=C \end{aligned}$$

下面考虑由边b_1，b_3，c_1，c_3，c_5组成的切割S。因为S包含树支b_1，b_3，所以S等于割集S_1和S_3的环和。其正确性可证明如下：

$$\begin{aligned} S_1\oplus S_3&=\{b_1,\ c_1,\ c_2\}\oplus\{b_3,\ c_2,\ c_3,\ c_5\} \\ &=S \end{aligned}$$

在前面的讨论中，我们证明了从一个生成树出发，可以构造出图的回路和割集子空间的基。这样构造出的基就是通常用于电网络研究中的基。

4.5　回路和割集子空间的关系

本节中，我们用图G的割集子空间子图的特性，得出G的回路子空间子图的一个特性。

在2.8节（定理2.14）中已证明回路和割集有偶数条公共边。因为图的回路子空间中的每个子图是一个回路或者是边不相交回路的并，而且割集子空间中的每个子图是一个割集或者是边不相交割集的并，所以下面得到了定理2.14的一个直接结果。

定理4.7　图G的回路子空间中的每个子图与G的割集子空间中的每个子图有偶数条公共边。　□

下面的定理将证明上述定理的逆定理。

定理4.8

1. 图G的子图属于G的回路子空间，如果它与G的割集子空间中的每个子图有偶数条公共边。

2. 图G的子图属于G的割集子空间，如果它与G的回路子空间中的每个子图有偶数条公共边。

证明

1. 不失一般性，假定G是连通的。当G不连通时，可按完全同样的方法去证明。

令T是G的一棵生成树，b_1，b_2，…表示T的树支，c_1，c_2，…表示T的连支。考虑与G的割集子空间中的每个子图有偶数条公共边的G的任一子图C。不失一般性，假定C包含连支c_1，c_2…c_r。令C'表示为对应于连支c_1，c_2，…，c_r的基本回路C_1，C_2，…，C_r的环和。

很显然，C'包含c_1，c_2，…，c_r，但不包含其它连支。因此$C'\oplus C$不包含连支。

G的某些回路的环和C'与G的割集子空间中的每个子图有偶数条公共边。因为C也具有这种性质，所以$C'\oplus C$也具有此性质。

现在来证明$C'\oplus C$是零。如果它不是零，则$C'\oplus C$仅包含树支。令b_i为$C'\oplus C$中任一树支。因此b_i是对应于b_i的基本割集与$C'\oplus C$之间的唯一公共边。因为$C'\oplus C$与每一割集必须有偶数条公共边，所以上述证明是不可能的。因此，$C'\oplus C$就应该是零。亦即$C=C'$

$=C_1\oplus C_2\oplus\cdots\oplus C_r$，而且 C 属于G的回路子空间。 □

2. 这部分可按同样方法来证明。

4.6 回路和割集子空间的正交性

由定理4.1可知，域F上每个n-维矢量空间与同一域上所有n-矢量的矢量空间同构。于是，G的矢量空间W_G与$GF(2)$上所有m-矢量的矢量空间同构，其中m是G中的边数。

令e_1，e_2，…，e_m表示G的m条边。假定我们给G的每个边导出子图G_i关联一个m-矢量w_i，并使当且仅当e_j在G_i中，w_i的第j个元素值等于1。那么，两个子图G_i和G_j的环和$G_i\oplus G_j$对应于m-矢量w_i+w_j，即w_i和w_j的模2加法。由此可见，刚才所描述的关联确实定义了W_G和$GF(2)$上所有m-矢量的矢量空间之间的同构。实际上，如果选$\{e_1\}$，$\{e_2\}$，…$\{e_m\}$作为W_G的基矢量，那么w_i的元素值就是G_i相对于这个基的坐标。

鉴于这种同构，我们还可以用W_G来表示与图G的子图相关联的所有m-矢量的矢量空间，用W_C来表示G的回路子空间中的子图的m-矢量的矢量空间，同样用W_S来表示G的割集子空间中的子图的m-矢量的矢量空间。

考虑任意两个矢量w_i和w_j，并使得w_i在W_C中，w_j在W_S中。因为W_C中每个子图和W_S中每个子图有偶数条公共边，所以w_i和w_j的点积$<w_i, w_j>$等于偶数个1的模2和，即$<w_i, w_j>=0$。换言之，W_C中的m-矢量和W_S中的m-矢量是相互正交的。于是有下面的定理。

定理4.9 图的回路和割集子空间是相互正交的。 □

下面考虑直接和$W_C⊞W_S$。我们知道

$\dim(W_C⊞W_S)=\dim(W_C)+\dim(W_S)-\dim(W_C\cap W_S)$

因为$\dim(W_C)+\dim(W_S)=m$，则得

$\dim(W_C⊞W_S)=m-\dim(W_C\cap W_S)$

当且仅当$\dim(W_C⊞W_S)=m$，正交子空间W_C和W_S也是W_G的正交互补空间。换言之，当且仅当$\dim(W_C\cap W_S)=0$，即$W_C\cap W_S$是所有元素均为零的零矢量，则W_C和W_S是正交互补空间。于是，得到下面的定理。

定理4.10 图的回路和割集子空间W_C和W_S是正交互补空间，当且仅当$W_C\cap W_S$是零矢量。 □

假定W_C和W_S是正交互补空间，则意味着W_G中每个矢量都可表示成w_i+w_j，其中w_i在W_C中，w_j在W_S中。换言之，G的每个子图都可表示成两个子图的环和，其中一个子图属于回路子空间，另一个属于割集子空间。特别是图G本身可表示成上述形式。

假定W_C和W_S不是正交互补空间。那么显然存在着一个子图，这个子图不能表示为W_C和W_S中子图的环和。在这种情况下产生的问题是：G能否用W_C和W_S的子图的环和来表示。回答是肯定的，如下面定理所述。

定理4.11 每个图G可表示成两个子图的环和，其中一个子图在G的回路子空间，另一个在G的割集子空间。 □

本定理的证明可见文献[4.4]和[4.5]。

举例来结束本节。

考虑图4.4(a)所示 G_a。可以证明，这个图中没有一个非零子图属于割集和回路子空间的交。于是 G_a的割集和回路子空间是正交互补空间。那么对应于 G_a某一个生成树的基本回路和基本割集集合构成了G_a矢量空间的一个基。一个对应于由边e_1，e_2，e_3和e_4组成的生成树的基本回路和基本割集集合给出如下：

基本割集矢量

$$S_1=(1\ 0\ 0\ 0\ 1\ 1\ 0)$$
$$S_2=(0\ 1\ 0\ 0\ 1\ 1\ 0)$$
$$S_3=(0\ 0\ 1\ 0\ 1\ 0\ 1)$$
$$S_4=(0\ 0\ 0\ 1\ 0\ 0\ 1)$$

基本回路矢量

$$C_1=(1\ 1\ 1\ 0\ 1\ 0\ 0)$$
$$C_2=(1\ 1\ 0\ 0\ 0\ 1\ 0)$$
$$C_3=(0\ 0\ 1\ 1\ 0\ 0\ 1)$$

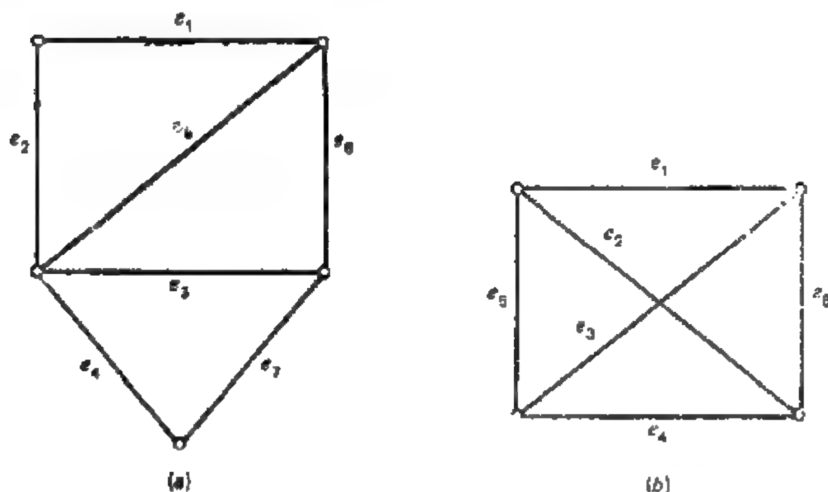


图 4.4 (a)图G_a；(b)图G_b。

现在可以证明，每个矢量都可以表示成一个回路矢量和一个割集矢量的环和。特别是代表G_a本身的矢量（1 1 1 1 1 1 1）可以表示成：

$$(1\ 1\ 1\ 1\ 1\ 1\ 1)=S_1\oplus S_2\oplus S_4\oplus C_1\oplus C_2$$
$$=(1\ 1\ 0\ 1\ 0\ 0\ 1)\oplus(0\ 0\ 1\ 0\ 1\ 1\ 0)$$

其中（1 1 0 1 0 0 1）代表G_a中的一个切割，而（0 0 1 0 1 1 0）代表 G_a 中的一个回路。

下面考虑图4.4(b)中的图G_b。图中e_1,e_2,e_3和e_4不仅组成了一个切割，而且还组成了一个回路。于是此图的回路和割集子空间不是正交互补空间。这意味着，G_b 中存在着某个不能用 G_b 的割集和回路子空间中子图的环和来表示的子图。然而，根据定理4.11，这样的分解对G_b是可行的。这也是对的。因为

确实有这样的子图。

$$G_b=\{e_1,\ e_2,\ e_5\}\oplus\{e_3,\ e_4,\ e_6\}$$

上式中$\{e_1,\ e_2,\ e_5\}$是G_b的一个切割，$\{e_3,\ e_4,\ e_6\}$是G_b的一个回路。

4.7 进一步阅读

文献[4.6]是有关图的矢量空间的早期论文，并且还讨论了构成具有给定回路集的图的问题。这个题目的进一步阅读文献还有文献[4.4]和[4.5]。

4.8 习　　题

4.1 证明：由下列边集所构成的回路组成了图4.5所示图的回路子空间的基：

$$\{e_1,\ e_3,\ e_4,\ e_6\},\ \{e_2,\ e_3,\ e_5,\ e_7\},\ \{e_1,\ e_2,\ e_8\}$$

4.2 一个顶点的关联集合就是与这个顶点相关联边的集合。证明：在n-顶点连通图中的任意$n-1$个关联集构成了此图的割集子空间的一个基。

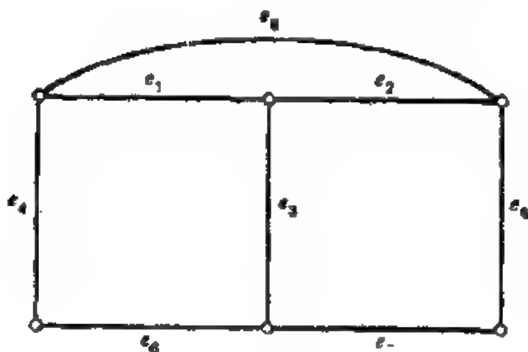
4.3 如果图G的秩是ρ，零度是μ，试证：

（ a ）G的割集子空间可能出现的不同基的数等于

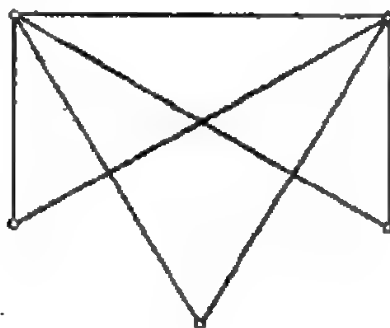
$$\frac{1}{\rho!}(2^\rho-2^0)(2^\rho-2^1)(2^\rho-2^2)\cdots(2^\rho-2^{\rho-1})$$

（ b ）G的回路子空间可能出现的不同基的数等于

$$\frac{1}{\mu!}(2^\mu-2^0)(2^\mu-2^1)(2^\mu-2^2)\cdots(2^\mu-2^{\mu-1})$$



刄 4.5


图 4.6

4.4 对于图4.6所示的图G，得出以下结论：

（ a ）回路子空间的一组基矢量，不是对应于G的任一生成树的基本回路矢量。

（ b ）割集子空间的一组基矢量，既不是对应于G的任一生成树的关联集合，也不是基本割集。

4.5 （ a ）检验如图4.6所示图G的割集和回路子空间是否为G的矢量空间的正交互补。

（ b ）用两个子图的环和来表示图G，一个子图来自回路子空间，另一个来自割集子空间。

4.6 证明：树的边集的每个子集都是树的一个切割。

4.7 证明：如果一个图的一个子图既属于图的割集子空间又属于回路子空间，那么它一定有偶数条边。

4.8 如果一个图的边的子集E'不包含回路，那么就说E'是独立的。证明下面的结论：

（ a ）独立集合的每个子集都是独立的。

（b）如果I和J分别是含有k和k+1条边的独立集合，那么必存在一条在J中但不在I中的边e，使得$I\cup\{e\}$是独立集合。

4.9 用割集来代替回路，重做习题4.8。

4.9 参考文献

4.1 S.Maclane and G.Birkhoff, *Algebra*, Macmillan, New York, 1967.

4.2 P.R.Halmos, *Finite-Dimensional Vector Spaces*, Van Nostrand Reinhold, New, York, 1958.

4.3 F.E. Hohn, *Elementary Matrix Algebra*, Macmillan, New York, 1958.

4.4 W.K.Chen, "On Vector Spaces Associated with a Graph," *SIAM J. Appl. Math.*, Vol.20, 526—529（1971）.

4.5 T.W.Williams and L.M.Maxwell, "The Decomposition of a Graph and the Introduction of a New Class of Subgraphs," *SIAM J. Appl. Math.*, Vol. 20, 385—389（1971）.

4.6 R.Gould, "Graphs and Vectors Spaces," *J. Math. Phys.*, Vol. 37, 193—214（1958）.

第五章 有 向 图

在前面四章，我们讨论了无向图理论的一些基本结论，但是有些情况却无法用无向图来表示。例如，在交通网络的图示中，每条边表示一条街道，而每条边需要给定一个方向以示出交通流的允许方向。又如，计算机程序可以用图来模拟，图中每条边表示从一组指令到另一组指令的控制流。在这种程序的表示中，需要给每条边指定方向以表示控制流的方向。电网络也是一个需要用有向图来表示物理系统的例子。在第十一章到第十五章中，将讨论有向图的应用和有关的算法。

本章将推导出有向图理论的几个基本结论，将讨论有向欧拉回路和哈密顿轨迹的存在性问题，还将讨论有向树以及与有向欧拉轨迹的关系。

5.1 基本定义和概念

首先引入几个与有向图有关的基本定义和概念。

有向图$G=(V, E)$含有两个集合：元素叫做顶点的有限集合V和元素叫做边的有限集合E。每条边与一个有序顶点对相关联。

我们用符号v_1，v_2，…来表示有向图的顶点，用符号e_1，e_2，…来表示有向图的边。如果$e_i=(v_i, v_j)$，那么v_i和v_j就叫做边e_i的端点，其中v_i叫做e_i的起点，v_j叫做e_i的终点，凡是具有相同起点和终点对的边叫做平行边。如果v_i既是某条边的起点又是它的终点，则这条边就叫做v_i点的自环。

在有向图的图示中，顶点用圆点或圆圈来表示。边用连接代表端点的圆点或圆圈的线段来表示。此外，每条边用由起点指向终点的箭头来标明其方向。

例如，如果

$$V=\{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$$

和

$$E=\{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\}$$

其中

$$e_1=(v_1, v_2)$$
$$e_2=(v_1, v_2)$$
$$e_3=(v_1, v_3)$$
$$e_4=(v_3, v_1)$$
$$e_5=(v_2, v_4)$$
$$e_6=(v_3, v_4)$$
$$e_7=(v_4, v_4)$$
$$e_8=(v_5, v_6)$$

那么，有向图$G=(V, E)$如图5.1所示。图中，e_1和e_2是平行边，e_7是自环。

每条边都说成与它的端点相关联。如果两个顶点是同一条边的端点，那么这两个顶点是邻接的；而如果两条边有一个公共端点，则这两条边也是邻接的。

每条边都是从它的起点射出，而射入它的终点。如果没有边与顶点关联，这个顶点就是孤立点。

顶点v_i的度$d(v_i)$是与v_i相关联的边数。v_i的入度$d^-(v_i)$是射入到v_i的边数，出度$d^+(v_i)$是由v_i射出的边数。δ^+和δ^-分别表示有向图中的最小出度和入度。而Δ^+和Δ^-分别表示最大出度和入度。

对于任意顶点v，集合$\Gamma^+(v)$和$\Gamma^-(v)$的定义如下：

$$\Gamma^+(v)=\{w\,|(v, w)\in E\}$$

$$\Gamma^-(v)=\{w\,|(w, v)\in E\}$$

例如，图5.1中的$\Gamma^+(v_1)=\{v_2, v_3\}$和$\Gamma^-(v_4)=\{v_2, v_3, v_4\}$。

注意，某一顶点上的自环对这个顶点的入度和出度的贡献都是一样的。每条边对有向图入度之和与出度之和的贡献都是1，下面的定理就是这个事实的结论。

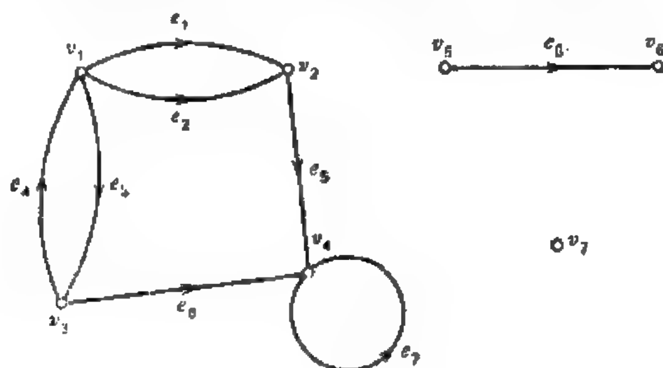


图 5.1　有向图

定理5.1　在有向图中，入度之和＝出度之和＝m，其中m是图的边数。□

有向图的子图和导出子图的定义与无向图的情况一样（1.2节）。

不考虑有向图G方向所得到的图叫做G的基础无向图，这个基础无向图记为G_u。

有向图$G(V, E)$中的一条有向通道是顶点$v_0, v_1, \cdots, v_k$的一个有限序列，而且(v_{i-1}, v_i)是G中一条边，$1\leqslant i\leqslant k$。这条通道通常叫做v_0—v_k有向通道，v_0和v_k叫做这条通道的起点和终点，其他顶点叫做内部顶点。一条有向通道的起点和终点叫做它的端点。注意，在一条有向通道中，边和顶点的出现可以不止一次。

如果一条有向通道的两个端点是不同的，那么这条有向通道是开的；否则就是闭的。

如果一条有向通道的边是不同的，则称为有向轨迹。如果一条有向轨迹的端点是不同的，则称为是开的；否则就是闭的。

如果一条开的有向轨迹的顶点是不同的，则称为有向路径。

如果一条闭的有向轨迹的顶点除端点外都不相同，则称为有向回路。

如果一个有向图不含回路，则称为无环的。如图5.2所示的图就是无环的。

如果一个顶点序列是基础无向图G_u的通道，它不一定是有向图 G 的有向通道[+]，如图5.2中，顶点序列v_1、v_2、v_3、v_4、v_5、v_3是一条通道，但不是一条有向通道。

在有向图中，轨迹、路径和回路的定义和无向图相类似。

如果一个有向图的基础无向图是连通的，那么这个有向图也是连通的。

如果有向图G的一个子图是G_u的一个片，那么这个子图也是 G的一个片。

如果有向图G中存在一条从顶点v_i到v_j的有向路径和一条从顶点v_j到v_i的有向路径，则称每个顶点对它本身是强连通顶点v_i对顶点v_j是强连通的。如果 v_i对v_j是强连通的，那么显然 v_j对 v_i也是强连通的。

如果v_i对v_j是强连通的；v_j对v_k是强连通的，那么容易看出，v_i对 v_k也是强连通的。因此在这种情况下，我们就简称v_i、v_j和v_k是强连通的。

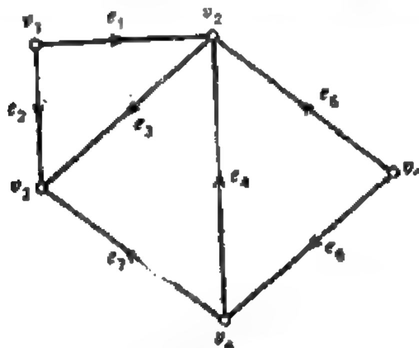


图 5.2　无环有向图

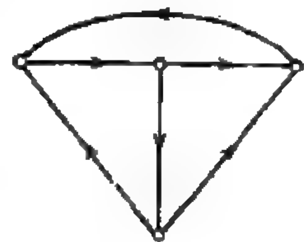


图 5.3　强连通有向图

一个有向图是强连通的，如果它所有的顶点都是强连通的。例如，图5.3所示的图是强连通的。

有向图G的最大强连通子图叫做G的强连通片。如果有向图是强连通的，那么它只有一个强连通片，即它本身。

考虑有向图$G=(V, E)$，容易看出，G的每个顶点仅出现在G的一个强连通片中，于是，强连通片的顶点集构成了图G顶点集V的一个划分。

例如，图5.4(a)的有向图有三个强连通片，即它们的顶点集$\{v_2, v_3, v_4, v_5\}$、$\{v_1\}$和$\{v_6\}$，这些顶点集构成了有向图G顶点集$\{v_1, v_2, v_3, v_4, v_5, v_6\}$的一个划分。

我们会有趣地注意到，有向图中某些边不属于任何强连通片。比如，在图5.4(a)中，边e_1、e_6、e_7、e_8和e_{10}就不在任何强连通片中。

这样，强连通性质虽然导出了顶点集的划分，但不一定能够导出边集的划分。

并、交、环和以及其它涉及到有向图的运算，其定义与无向图情况下的定义完全相同（见1.5节）。

把有向图G中每个强连通片的所有边都收缩后所得到的图叫做G的边缩图G_c。例如，图5.4(a)的边缩图如图5.4（b）所示。

G_c中对应于强连通片的顶点叫做这个片的缩象。

有向图的秩和零度与所对应无向图的相同。这样，如果有向图G有m条边，n个顶点，

[+] 对原文作了部分改正。——译者注

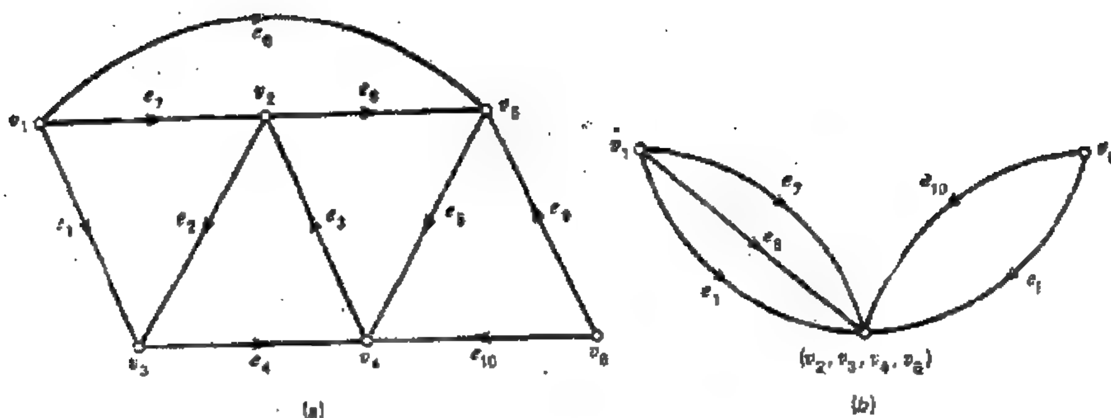


图 5.4 一个图和它的边输图

p 个片，那么G的秩 ρ 和零度 μ 可由下式给出

$$\rho = n - p$$

和

$$\mu = m - n + p$$

下面来定义最小连通有向图，并研究它的某些性质。

如果G是强连通的，而从G中取走任一条边都将破坏图的"强连通"性，则称有向连通图 G 为最小连通。例如，图 5.5 就是最小连通的。

显然，最小连通有向图既不能有并行边，又不能有自环。

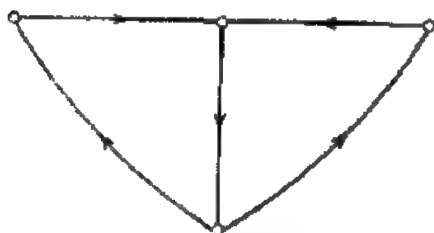


图 5.5 最小连通有向图

我们知道，无向图是最小连通的，当且仅当它是一个树（习题 2.13）。由定理 2.5可知，一个树至少有两个顶点的度数为 1 。这样，最小连通无向图也至少有两个顶点的度数为 1 。

现在，对有向图推导出相似的结论。在强连通有向图中，因为在每个顶点上至少有一条边射入和一条边射出，所以每个顶点的度数至少为 2 。下面的定理将证明，在最小连通有向图中至少有两个顶点的度数为 2 。

定理 5.2 若最小连通有向图G具有一个以上顶点，那么G至少有两个度数为 2 的顶点。

证明

因为G是强连通的，并且含有一个以上顶点，所以它至少有一个有向回路。这样，G的零度 $\mu \geqslant 1$。

我们将通过对 μ 的归纳来证明定理5.2。

如果 $\mu = 1$，那么G一定是一个有向回路，因此定理5.2在这种情况下成立。

假设定理5.2对所有零度 $\mu < k$ 的最小连通有向图成立，$k > 2$。现在考虑 $\mu = k$ 的最小连通图 G，我们将证明定理5.2对G也成立。

下面分两种情况。

情况 1 G中每个有向回路的长度为 2 。

在这种情况下，G的任意两个相邻顶点仅由两条不同方向的边相连接。令G'是与G有相同顶点集的简单无向图，其中两个顶点是邻接的，当且仅当它们在G中是邻接的。因为G是连通的，所以G'也是连通的。另外，因为G不含有长度大于 2 的回路，所以G'不含有回路。这样，G'是一个树。因此由定理2.5可知，G'至少有两个悬挂点。这两个顶点在G中的度数为 2 ，因而定理对这种情况成立。

情况 2　G有一个长度$l \geq 3$的回路C。

因为G是最小连通的，所以G中C的任意两个相邻顶点间仅有一条边，不相邻顶点间却没有边。

假设G'是G通过收缩C中的边而得到的图。G'有$(m-l)$条边和$(n-l+1)$个顶点，其中m、n分别是G中边数和顶点数。因此，G'的零度等于

$$(m-l)-(n-l+1)=k-1$$

因为G'是最小连通的，由归纳假设可知，它至少有两个度数为 2 的顶点v_1和v_2。如果其中有一个顶点（比如v_1）是C的缩象，那么C中至少有一个顶点，比如说是v_3，其度数为 2 。此时，v_2和v_3是G中两个度数为 2 的顶点。

另一方面，如果v_1和v_2都不是C的缩象，那么v_1和v_2就是G中两个度数为 2 的顶点。

于是定理证毕。 □

本节最后来定义准强连通性质。

在一个有向图中，如果对于每对顶点v_1和v_2，存在顶点v_3，并从v_3到v_1和v_2各有一条有向路径，那么这个有向图称为是准强连通的。注意，v_3不必与v_1、v_2不同。

例如，图 5.6 中的图就是准强连通的。

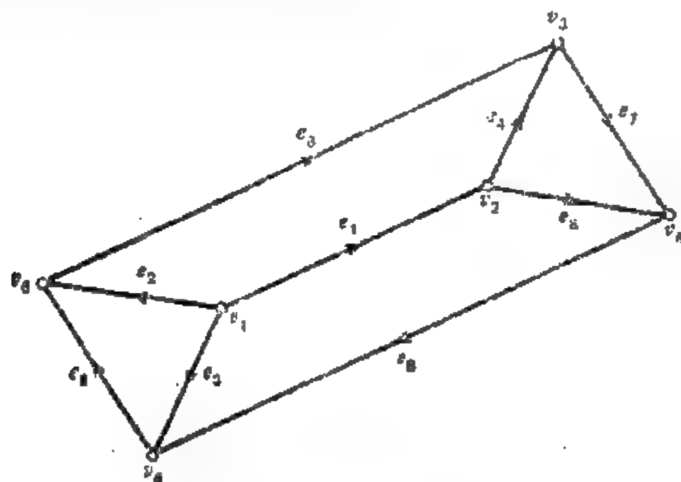


图 5.6 准强连通图

5.2 图和关系

集合$X=\{x_1, x_2, \cdots\}$上的二元关系是X的有序元素对的组合。例如，如果集合X是由男人组成的，R是表示"…是…的儿子"关系，那么有序对(x_i, x_j)就意味着x_i是x_j的儿

子，也可记为$x_i R x_j$。

一种最普通的方法是用有向图来表示集合x上的二元关系，其中顶点代表X的元素，边代表定义关系R的x的有序元素对。

例如，集合$X=\{2,3,4,6,9\}$上"…是…的因子"关系，如图5.7所示。

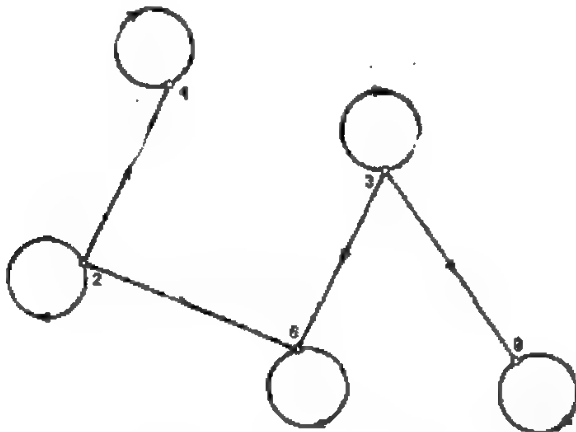


图 5.7 表示"…是…的因子"关系的有向图

考虑集合$X=\{x_1, x_2, \cdots\}$和X上的关系R。

1.如果每个元素x_i在关系R中对应于它自身，则称R是反射的；即对于每个x_i，有$x_i R x_i$。

2.如果$x_i R x_j$包含$x_j R x_i$，则称R是对称的。

3.如果$x_i R x_j$和$x_j R x_k$包含了$x_i R x_k$，则称R是传递的。

4.如果R是反射的、对称的、传递的，则R是等价关系。如果R是定义在集合S上的等价关系，那么就可以把S唯一地分成子集$S_1, S_2, \cdots, S_k$，使得S的元素x和y属于S_i，当且仅当xRy。子集S_1，$S_2, \cdots, S_k$叫做由集合S上的关系R导出的等价族。

假定集合X由正整数组成，那么

1."是…的因子"关系是反射的和传递的。

2."等于"关系是反射的、对称的和传递的，因而也是等价关系。

表示"反射关系"的有向图叫做反射有向图。对称有向图和传递有向图也可类似地定义。我们对这些图做下列观察：

1.在反射有向图中，每个顶点有个自环。

2.在对称有向图中，任意两个相邻顶点间存在着两条方向相反的边。因而，如果把一条边想象为两条反向边，就可以用无向图来表示对称关系。

3.如果G中有一条从v_1到v_2的有向路径，那么边（v_1，v_2）就出现在传递图G中。

5.3 有向树或单向树

在有向图G中，如果从顶点v到G的其他所有顶点都存在有向路，则顶点v称为G的根。

例如，在图5.6的图G中，顶点v_1是一个根。很显然，如果某个图含有一个根，那它就

是准强连通的。在下面的定理中，我们将证明，“准强连通”性质包含着根的存在。

定理5.3 有向图G有一个根，当且仅当它是准强连通的。

证明

必要性 这是很显然地。

充分性 考虑G的顶点x_1、x_2、…x_n。因为G是准强连通的，所以存在顶点y_1，从它到x_1和x_2各有一条有向路径。由于同样原因，还存在顶点y_2，从它到y_1和x_3也各有一条有向路径。很显然，y_2也是通过y_1的有向路径连接到x_1和x_2的。如此进行下去，可看出，存在顶点y_n，并从它到y_{n-1}和x_n各有一条有向路径。因为y_n是通过y_{n-1}连接到x_1、…、x_{n-1}的，显然y_n是一个根。 □

如果有向图G的基础无向图是一个树，则有向图G也是一个树。

如果有向图G是树且含有一个根，则有向图G是有向树或单向树。如果$d^+(v)=0$，则G中的一个顶点v叫做叶。

例如，图5.8的图是一个有向树。图中v_1的根，而且是唯一的根。

下面的定理将介绍有向树的一些等价特性。

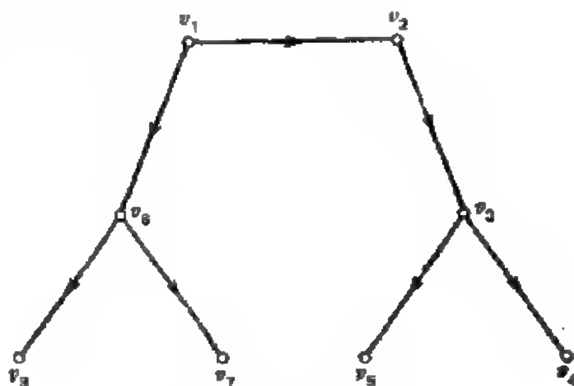


图 5.8 有向树

定理5.4 令G是具有$n>1$个顶点的有向图。因此，下列的论点是等价的：

1. G是一个有向树。

2. G中存在一个顶点r，而且从r到G的其它每个顶点恰好有一条有向路径。

3. G是准强连通的，如果从它移去任何一条边，就会失去这种性质

4. G是准强连通的，并含有一个顶点r，使得

$$d^-(r)=0$$

和

$$d^-(v)=1, v\neq r$$

5. G不含有回路（并不一定是有向回路），但是有一个顶点r，使得

$$d^-(r)=0$$

和

$$d^-(v)=1, v\neq r$$

6. G是无回路的准强连通图。

证明

$1\Longrightarrow2$ G中的根r具备所要求的性质。

$2\Longrightarrow3$ 显然G是准强连通的。假若从G中移去边(u,v)时，这个性质不被破坏。那么存在顶点z，使得从z到x和y有两条有向路径，并未用到边(u,v)。这样，G中存在两条从

x到v的有向路径，从而存在两条由r到v的有向路径。这与论点2相矛盾。

$3\Longrightarrow4$　因为G是准强连通的，它有一个根r。所以，对每个顶点$v\neq r$，

$$d^-(v)\geqslant1$$

假定，对某个顶点v，有

$$d^-(v)>1$$

那么就有两条路径（比如说(x,v)和(y,v)）射入v。这样，从r到v存在着两条有向路径，一条经过(x,v)，另一条经过(y,v)。所以，既使我们移去其中的一条边，生成的图仍然是以r为根，而且还是准强连通的。这与论点3相矛盾，所以，

$$d^-(v)=1,\qquad v\neq r$$

最后，如若没有边射入到r，则移去G中这条边，这样生成的图仍然是以r为根，而且还是准强连通的。这仍与论点3相矛盾，故

$$d^-(r)=0$$

$4\Longrightarrow5$　G的入度之和等于$n-1$。由定理5.1可知，G有$n-1$条边。因为G仍是连通时，由定理2.1的论点3可知，G是一个树，因而G是无回路的。

$5\Longrightarrow6$　因为G的入度之和等于$n-1$，所以G有$n-1$条边。又因G是无回路的，由定理2.2可知，它是一个树。因此，G中存在一条从r到其它每个顶点的唯一路径。这条路径应该是有向路径，否则，至少有一个顶点的入度大于1，这与论点5相矛盾。这样，顶点r是G的一个根。因此，由定理5.3可知，G是准强连通的。

$6\Longrightarrow1$　因为G是准强连通的，所以由定理5.3可知，它有一个根。进而，因为G是连通和无回路的，所以它是一个树。□

有向图G的一个子图是G的一个有向生成树，如果这个子图是一个有向树且包含了G的所有顶点。例如，由边e_1、e_2、e_3、e_4和e_5组成的子图就是图5.6所示图的生成树。

我们知道图G有一个生成树，当且仅当G是连通的。在有向图情况下，一个相应的定理如下。

定理5.5　有向图G有一个有向生成树，当且仅当G是准强连通的。

证明

必要性　如果G有一个有向生成树，那么这个有向树的根显然也是G的根。因此，由定理5.3可知，G是准强连通的。

充分性　假定G是准强连通的，但不是有向树。那么由定理5.4的论点3可知，从G中移去这些边，不会破坏G的准强连通性。因此，只要我们连续地从G中移去所有的这些边，那么所得到的图就是G的一个有向生成树。□

有向树在有源网络拓扑研究中的应用将在第十三章讨论。

二元树是一类有向树，其中每个顶点的出度最多是2。例如，图5.9所示图就是二元树。在计算机科学的某些研究，例如，算法分析，搜索技术等等中，二元树是非常有用的。下面叙述这方面的一个问题。

若给定n个权w_1，w_2，…，w_n和n个长度l_1，l_2，…，l_n，需构造出具有n个叶v_1，v_2，…，v_n的二元树，并使

1. w_i是顶点的v_i的权，$1 \leqslant i \leqslant n$。

2. l_i是由根到顶点v_i的路长。$1 \leqslant i \leqslant n$

3. 称为加权路长的 $\sum_{i=1}^{n} w_i l_i$是最小的。

下面叙述一种比上面问题更加普遍的情况。

在一组字中，任何一个字都不是其它字的开关，这组字叫做前缀码。很明显，如果一个字母序列是由前缀码串接而成的，那么从左到右读这个序列，就可以把它分解为一个个单独的前缀码字。例如，{000,001, 01, 10, 11}是前缀码，由这种码字串接而成的序列1011001000，就很容易分解为字10，11，001和000。

假定$S=\{0, 1, 2, \cdots, m-1\}$是$m$个文字的字母表。令$T$是有向树，且有

1. 每个顶点的出度最多为m;

2. 每个顶点的每条射出边与字母表S中的一个文字相对应，以防两条这样的边与同一文字相对应。

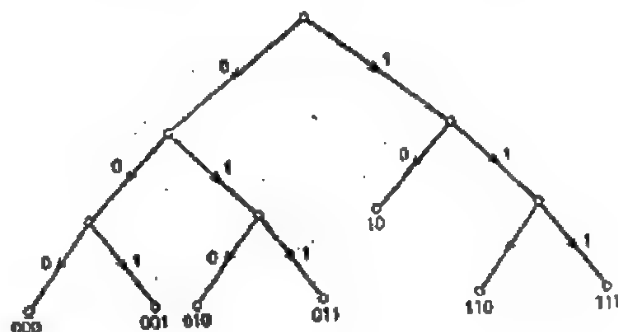


图 5.9 二元树

那么每片叶v就与一个字相对应，这个字是由顺序串接所有从根到v的路边上所出现的文字而成。还可以证明，对应于叶的字组成了前缀码。例如，对应于图5.9有向树的前缀码为

$$\{000, 001, 010, 011, 10, 110, 111\}$$

现在的问题是：已知m和长度$l_1, l_2, \cdots, l_n$，如何用字母表$S=\{0, 1, 2, \cdots, m-1\}$来构造出码字长度为$l_1, l_2, \cdots, l_n$的前缀码。

这个问题以及某些有关的问题，将在15.2节和15.3节中讨论。

5.4 有向欧拉图

有向图G的有向欧拉轨迹是包含了G的所有边的闭合有向轨迹。有向欧拉开轨迹是包含了G中所有边的有向开轨迹。含有向欧拉轨迹的有向图叫做有向欧拉图。

图5.10所示图G是有向欧拉图，因为边序列$e_1, e_2, e_3, e_4, e_5, e_6$组成了$G$中的一条有向欧拉轨迹。

下面的定理给出了有向欧拉图的简捷实用的性质。证明定理的思路与证明定理3.1的相同。

定理5.6 对于连通有向图G，下列论点是等价的：

1. G是有向欧拉图。

2. 对于G的每个顶点v，有$d^-(v)=d^+(v)$。

3. G是某些边不相交的有向回路的并。 □

例如，考察图5.10所示有向欧拉图G。容易证明，它满足上述定理的性质2，并且还是边不相交回路$\{e_2, e_3\}$和$\{e_1, e_4, e_5, e_6\}$的并。

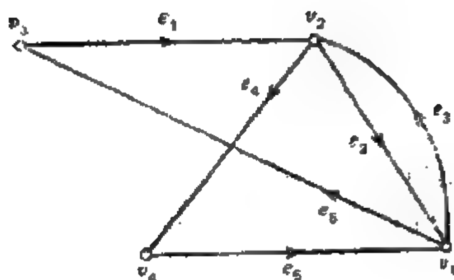


图 5.10　有向欧拉图

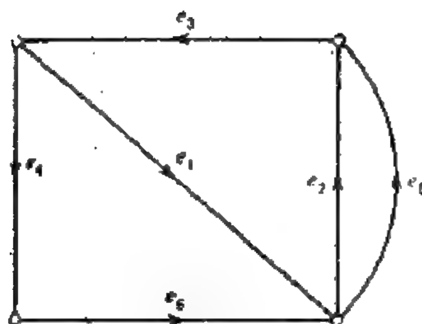


图 5.11　有向欧拉开轨迹的图

下面的定理很容易证明。

定理5.7　一个连通有向图具有一条有向欧拉开轨迹，当且仅当它满足下列条件：

1. G中存在两个顶点v_1和v_2，而且

$$d^+(v_1)=d^-(v_1)+1$$

和

$$d^-(v_2)=d^+(v_2)+1$$

2. 对于每个不同于v_1和v_2的顶点v，有

$$d^-(v)=d^+(v)$$ □

例如，图5.11所示图G满足上述定理的条件。边序列e_1，e_2，e_3，e_4，e_5，e_6是G中的一条有向欧拉开轨迹。

下面来讨论定理5.6的一个有趣的应用。

令$S=\{0, 1, \cdots, s-1\}$是有s个文字的字母表。显然，我们可以用字母表S中的文字，来构造出s^n个长度为n的不同的字。de Bruijn序列是长度为$L=s^n$的循环序列$a_0, a_1, a_2, \cdots, a_{L-1}$，而且对于每一个长度为$n$的字$w$，总存在唯一的$i$，使得

$$w=a_i a_{i+1} \cdots a_{i+n-1}$$

其中，下标是进行模L的计算。de Bruijn序列在编码理论和通讯电路的研究中得到了应用。更多有关这些序列的信息，可参阅文献[5.1]和[5.2]。现在我们要考虑的问题是：

对于每个$s\geqslant 2$和任何整数n，是否在字母表$S=\{0, 1, 2, \cdots, s-1\}$上存在一个de Bruijn序列？

我们将会看到，对上面问题的回答是肯定的。

我们将证明，对每个$s\geqslant 2$和每个n，总存在一个有向欧拉图$G_{s,n}^*$，使得图中每条有向欧拉轨迹都对应于字母表$S=\{0, 1, 2, \cdots, s-1\}$上长度为$s^n$的一个de Bruijn序列。图$G_{s,n}^*=(V, E)$按下面的步骤构造：

1. V由字母表S上长度为$n-1$的s^{n-1}个字组成。

2. E是S上长度为n的s^n个字的集合。

3. 边$b_1b_2\cdots b_n$是以$b_1b_2\cdots b_{n-1}$为起点，以$b_2b_3\cdots b_n$为终点。注意，每个顶点都有s条射入边和s条射出边。于是，对每一个顶点v，有$d^-(v)=d^+(v)$。

例如，图$G_{2,4}^*$和$G_{2,3}^*$如图5.12(a)和(b)所示。

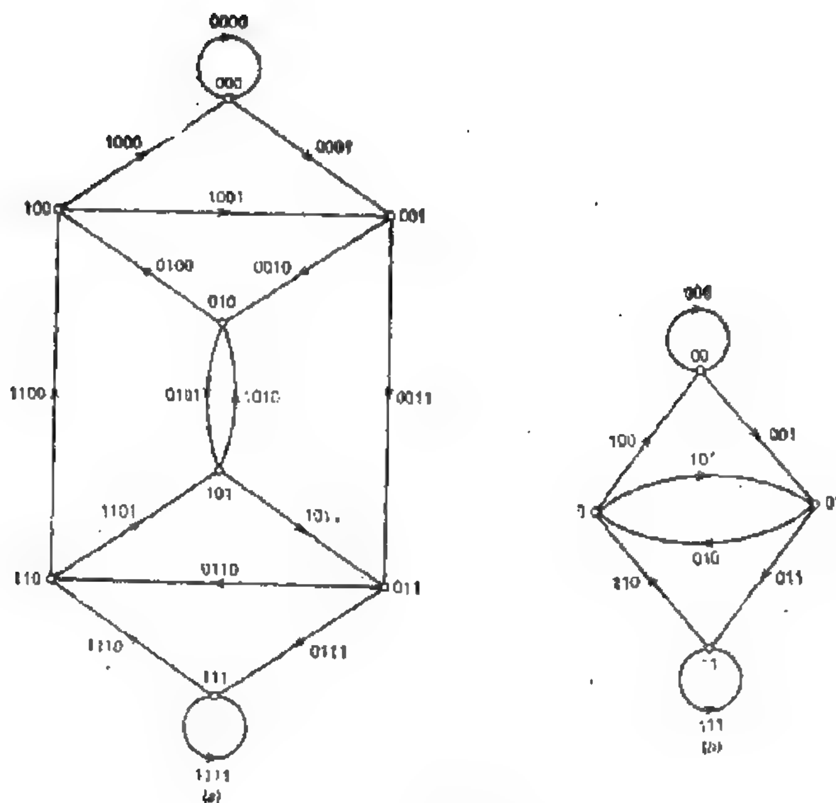


图 5.12 (a)$G_{2,4}$ (b)$G_{2,3}$

假定$G_{s,n}^{*}$中存在一条有向欧拉轨迹C。如果我们把代表$G_{s,n}^{*}$边的每个打头字母，按照它们在C中出现的顺序串接起来，那么就得到了长度为s^n的序列，从而使得此序列的每个具有n个连续字母的子序列对应于一个唯一的字，并且不存在两个不同的子序列对应于同一个字。这样，如上所构造的序列就是de Bruijn序列。例如，边序列000，001，011，111，110，101，010，100是$G_{2,3}^{*}$中的一条有向欧拉轨迹。把这些字的打头字母串接起来，就得到了de Bruijn序列00011101。

因此，为了说明对于每个$s\geqslant 2$和每个n都存在一个de Bruijn序列，只需证明$G_{s,n}^{*}$是一个有向欧拉图。证明在下面的定理中给出。

定理5.8 $G_{s,n}^{*}$是有向欧拉图。

证明

因为对于任意两个顶点$v_1=b_1b_2\cdots b_{n-1}$和$v_2=c_1c_2\cdots c_{n-1}$，总存在一条从v_1到v_2的有向路径，此路径由边$b_1b_2\cdots b_{n-1}c_1$，$b_2b_3\cdots b_{n-1}c_1c_2$，…，$b_{n-1}c_1c_2\cdots c_{n-1}$组成所以$G_{s,n}^{*}$是连通的。正如前面所见，因为$G_{s,n}^{*}$中每个顶点的入度等于其出度，由定理5.6可知，$G_{s,n}^{*}$是有向欧拉图。 □

推论5.8.1 对于每个$s\geqslant 2$和每个n，总存在一个de Bruijn序列。 □

5.5 有向生成树和有向欧拉轨迹

设G是无自环的有向欧拉图。本节将给出G中欧拉轨迹数目和有向生成树数目之间的

关系。

令v_1，v_2，…，v_n表示G的顶点。考虑G中一条有向欧拉轨迹C。令e_{i_1}是G的任一条与v_1相关联的边。对于每个$p=2$，3，…，n，令e_{i_p}表示经过e_{i_1}后在C上第一次进入顶点v_p的边。例如，在图5.10所示的有向欧拉图中，边序列e_1，e_2，e_3，e_4，e_5，e_6是一条有向欧拉轨迹。如果我们选择$e_{i_1}=e_6$，那么$e_{i_2}=e_1$，$e_{i_3}=e_6$以及$e_{i_4}=e_4$。

令H表示G在边集$\{e_{i_2}, e_{i_3}, \cdots, e_{i_n}\}$上的子图。

引理5.1. 令C是有向欧拉图G的一条有向欧拉轨迹，由上面定义的子图H是G的一棵以v_1为根的有向生成树。

证明

在H中，对于所有$p=2$，3，…，n，显然有$d^-(v_1)=0$以及$d^-(v_p)=1$。假定H有一个回路C'。那么v_1不在C'上，否则对于C'上其它顶点v，有$d^-(v_1)>0$或$d^-(v)>1$。由于同样的原因，C'是有向回路。因为对于C'上每个顶点，$d^-(v)=1$，所以不在C'中的边不会进入C'中的顶点。这意味着，当边e是从C经过e_{i_1}后第一条射入到C'的顶点的边时，边e就不属于H，这与H的定义相矛盾。于是H不含有回路。

由定理5.4.的论点5可知，H是G的有向生成树。 □

给定具有n个顶点且无自环的有向欧拉图的一个有向生成树H，令v_1是H的根，e_{i_1}是G中射入到v_1的边。对于$p=2$，3，…，n，令e_{i_p}是H中进入v_p的边。现在叙述一种在G中构造一条有向欧拉轨迹的方法。

1. 从顶点v_1出发，反向经过除e_{i_1}外的任一条射入v_1的边；如果没有这样的边，就经过e_{i_1}。

2. 一般情况下，当到达顶点v_p后，如果可能的话，则离开v_p时，反向经过除e_{i_p}外的一条射入v_p而未曾经过的边。如果不存在这样的边，则停止。

在上面的程序中，我们每次都到达一个顶点$v_p\neq v_1$。因为G中每个顶点的入度等于出度，所以必然存在一条曾未经过且指向v_p的边。因此，这个程序仅仅是在经过所有射入到v_1的边后，终止在v_1上。

假定当上面的程序终止在v_1时，G中就存在一条曾未经过的边(u, v)。因为u的入度等于其出度，则至少存在一条射入到u的曾未经过的边。如果有不止一条这样曾未经过的边，那么其中必有一条从H中指向u的边y，这可由上述程序的步骤2得出。这条曾未经过的边y将引出另一条在H中曾未经过的边。最后，我们将到达v_1，并找到一条射入到v_1的曾未经过的边，因为当程序终止在v_1时，所有射入到v_1的边都已经过这种选择是不可能的。

这样，在上述的程序中，G中所有的边都将经过，从而确实构造出了一个有向欧拉轨迹。

因为每个顶点v_p都有$(d^-(v_p)-1)!$种选择入射边（e_{i_p}在末端）的不同顺序，所以由一个给定的有向生成树H和e_{i_1}所构造出的不同的有向欧拉轨迹数目是

$$\prod_{p=1}^{n}(d^-(v_p)-1)!$$

此外，对于某些$p=2, 3, \cdots, n$；H的每种不同的选择都将产生一个不同的e_{i_p}。从而在最后的有向欧拉轨迹中，经过e_{i_1}之后，从不同的边射入到v_p。

最后，因为构造有向欧拉轨迹的过程是构造有向生成树的逆过程，因此每个有向欧拉轨迹都可由某个有向生成树构造而成。

这样，我们从文献[5.3]已经证明了下述定理。

定理5.9. 无自环有向欧拉图G的有向欧拉轨迹的数目是

$$\tau_d(G)\prod_{p=1}^{n}(d^-(v_p)-1)!$$

其中$\tau_d(G)$是G的以v_1为根的有向生成树的数目。 □

因为有向欧拉轨迹的数目与根的选择无关，因而得到下面推论。

推论5.9.1 有向欧拉图的有向生成树的数目对于根的每种选择都是相同的。 □

在6.9节中，我们将导出计算$\tau_d(G)$的公式。

5.6 有向哈密顿图

如果有向图G中的一条有向回路含有G的全部顶点，那它就是G的一个有向哈密顿回路，如果G中的一条有向路含有G的全部顶点，那它就是G的一条有向哈密顿路。

一个图是有向哈密顿图，如果它有一条有向哈密顿回路。

例如，在图5.13(a)的图中，边序列e_1、e_2、e_3、e_4、e_5、e_6是有向哈密顿回路，在图5.13(b)中，边序列e_1，e_2，e_3，e_4，e_5是有向哈密顿路。注意，后面这个图没有有向哈密顿回路。

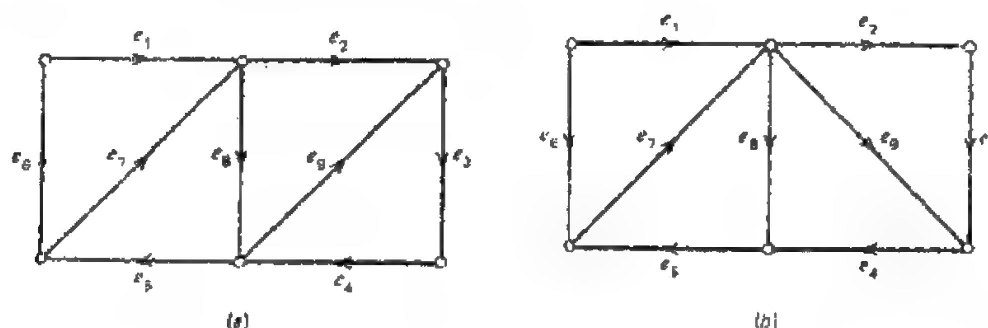


图 5.13 (a)有向哈密顿图；(b)有向图中含有哈密顿路而无哈密顿回路。

表征一个有向哈密顿图与表征无向哈密顿图是同样的困难。但是，有几个充分条件可以保证有向哈密顿回路和路的存在。现在就来讨论其中的几个。

有向图G是完备的，如果它的基础无向图是完备的。

下面的定理由文献[5.4]给出。

定理5.10 令u是具有n个顶点的强连通完备有向图的任一个顶点，$n\geqslant 3$。对于每一个k，$3\leqslant k\leqslant n$，存在一条包含u且长度为k的有向回路。

证明

令$G=(V,E)$是具有n个顶点的强连通完备有向图，$n\geqslant 3$。令$S=\Gamma^+(u)$和$T=\Gamma^-$

(u)。因为G是强连通的，所以S和T都是非零的。由于同样的原因，存在一条从$v\in S$到$w\in T$的有向路径。这样，u是在一个长度为3的有向回路中（见图5.14）。

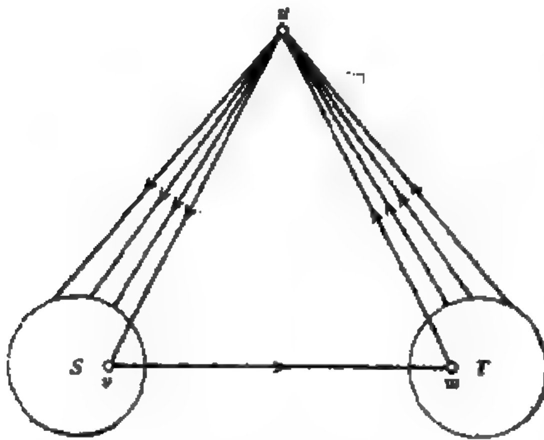


图 5.14

我们通过对k的归纳来证明定理。假定u是在全长为3到p的有向回路中，其中$p<n$。现在来说明，u是在长度为$p+1$的有向回路中。

令$C=u=v_0, v_1, v_2, \cdots, v_p=u$是长度为$p$的有向回路。这里需要考虑两种情况。

假定，对于某个不在C上的顶点v，存在着两条有向边(x, v)和(v, y)，其中x和y在C上。那么，在C上存在着两个相邻顶点v_i和v_{i+1}，使得有向边(v_i, v)和(v, v_{i+1})属于G。因此，在这种情况下，u是在长度为$p+1$的有向回路$u=v_0, v_1, v_2, \cdots, v_i, v, v_{i+1}, \cdots, v_p=u$上。

否则，令S为那些不在C中的顶点集合，它们是从C中顶点射出边的终点，再令T为那些不在C中的顶点集合，它们是射向C中顶点边的起点。又因为G是强连通的，所以S和T都是非零的。此外，还存在一条由顶点$v\in S$指向$w\in T$的边。所以u是在长度为$p+1$的有向回路$u=v_0, w, v, v_2, \cdots, v_p=u$中（见图5.15）。 □

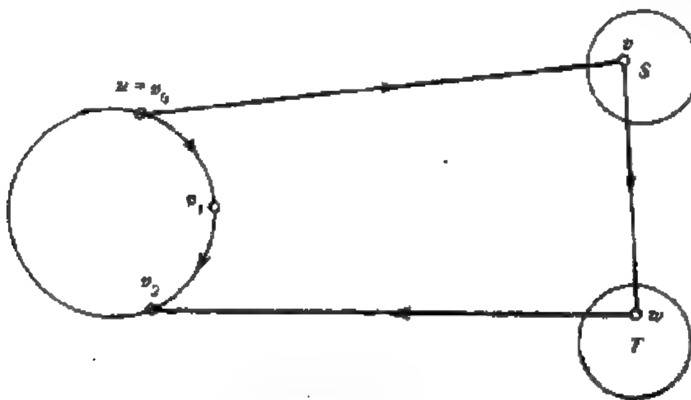


图 5.15

推论5.10.1 强连通完备有向图是哈密顿图。 □

下面来叙述Ghouila-Houri给出的极为有效的定理，但不加证明。这个定理的证明是十分繁琐的，可在文献[5.3]（pp.196—199）中找到。

定理5.11 令G是具有n个顶点无自环和无并行边的强连通有向图。如果对于G中的每个顶点v，有

$$d^-(v)+d^+(v)\geq n$$

那么，G就含有一个有向哈密顿回路。 □

下面是对Dirac的无向图结论（推论3.4.1）的推广，它可用定理5.11和习题5.6来证明。

推论5.11.1 令G是具有n个顶点无并行边或无自环的有向图。如果$\min\{\delta^-, \delta^+\}\geq \frac{1}{2}n\geq 1$，那么$G$就包含一个有向哈密顿回路。 □

这个结论的直接证明可参阅文献[5.6]（pp.178—179）。

结束本节之前，将给出有向哈密顿路存在的结论。

定理5.12 如果有向图$G(v, E)$是完备的，那么它就包含一条有向哈密顿路。

证明

考虑G中最长有向路P：a_1，a_2，…，a_p。假定b不是P上的顶点。因此，边$(b, a_1)\notin E$，否则，路径b，a_1，a_2，…a_p将比P更长。于是，$(a_1, b)\in E$。

若现在$(b, a_2)\notin E$，否则，a_1，b，a_2，…，a_p将比P有更长的路径。于是，$(a_2, b)\in E$。

重复上面的讨论，我们发现$(a_p, b)\in E$。但是对于a_1、a_2、…，a_p，这与b比P有更长的路径是矛盾的。因此，在P以外没有顶点。因而，P是G中一条有向哈密顿路。 □

5.7 无圈有向图

本节中，我们研究一类重要的有向图的性质，即无圈有向图。众所周知，有向图是无圈的，如果它不包含有向回路。显然，无圈有向图最简单的例子就是有向树。

本节的主要结论是，我们能够用集合$\{1,2,\cdots,n\}$中的整数来标记n个顶点无圈有向图G的顶点，这样出现的边(i, j)暗指$i<j$。注意边(i, j)是从顶点i指向顶点j。对顶点进行上述有序的标号叫做拓扑分类。例如，图5.16的无圈有向图就是拓扑分类的，本节主要结果的证明依赖于下述定理。

定理5.13 在无圈有向图G中，至少存在一个入度为零的顶点和一个出度为零的顶点。

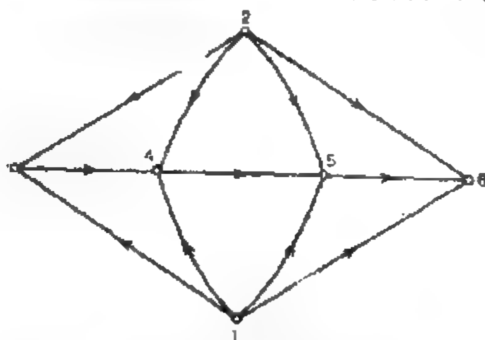


图 5.16 无圈有向图

证明

令P：v_1，v_2，…，v_p是G中一条最长的有向路径。我们要求v_1的入度和v_p的出度都等于零。

如果v_1的入度不为零，那么存在一个顶点w，使得边（w，v_1）在G中。现在来检查两种情形。

情形1 假定对于任何i，$w\neq v_i$，$1\leq i\leq p$。那么存在一条包含了P中所有边的有向路径P'：w，v_1，…，v_p。这与P是最长有向路径相矛盾。

情形2 假定对于某个i，$w=v_i$，那么在G中存在一个有向回路C：v_1，v_2，…，v_i，v_1。因为G不含有向回路，这又产生了矛盾。

于是，不存在顶点w，使得（w，v_1）在G中，即v_1的入度为零。

按照相似的推论过程可以证明，v_p的出度为零。 □

下面着手对n个顶点无圈有向图的顶点进行拓扑分类。

选取任意一个出度为零的顶点。因为G是无圈的，那么由定理5.13可知，G中至少存在一个出度为零的顶点，并把这个顶点标记为n，若从G中移去这个顶点以及射入它的边便得图G'。因为G'也是无圈的，所以可选出G'中一个出度为零的顶点，并把它标记为$n-1$，重复上述步骤，直到所有顶点标记完为止。容易证明，这个过程导出了G的顶点的一个拓扑分类。

例如，图5.16的顶点标号就是按照上述过程进行的。

5.8 比赛图

比赛图是完备有向图。由于比赛图是用来表示循环比赛的组织，因而得名。在一次循环比赛中，有若干个队参加，每个队与其它各个队正好比赛一次，每场比赛不能出现平局。在表示循环比赛的有向图中，顶点表示各队，如果顶点v_1代表的队击败顶点v_2代表的队，那么就在图中出现一个边（v_1，v_2）。显然，这样得到的有向图没有并行边和自环，但在任意两个顶点间恰好有一条边。因此，它是一个完备有向图，也就是一个比赛图。图5.17示出了一个比赛图。

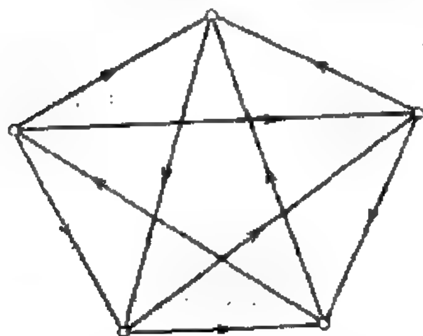


图 5.17 比赛图

参加比赛的队可以按照得分来排列。一个队的得分是它所击败的队的数目。这就引出比赛图得分序列的定义。

n个顶点比赛图的得分序列是序列（s_1，s_2，…，s_n），其中每个s_i是比赛图中一个顶点的出度。在下面的定理中，给出用得分序列来表示比赛图的一个有趣性质。

定理5.14 一个非负整数序列s_1，s_2，…，s_n是比赛图G的得分序列，当且仅当

1. $\sum_{i=1}^{n} s_i=\frac{n(n-1)}{2}$。

2. $\sum_{i=1}^{k} s_i \geqslant \frac{k(k-1)}{2}$，　　对于所有$k<n$。

证明

必要性　注意，求和式$\sum_{i=1}^{n} s_i$等于比赛图G中的边数（定理5.1）。因为比赛图是完备图，所以它有$n(n-1)/2$条边，其中n是顶点数。因而

$$\sum_{i=1}^{n} s_i=\frac{n(n-1)}{2}$$

为了证明条件2，考虑任意k个顶点v_1、v_2、…、v_k上的子比赛图。这个子比赛图有$k(k-1)/2$条边。因为可能存在由子比赛图中顶点指向子比赛图外顶点的边，因此，在整个比赛图中

$$\sum_{i=1}^{k} s_i \geqslant \frac{k(k-1)}{2}$$

充分性　见文献[5.5]（pp.107—109）　□

假定我们能够对一场循环比赛中的各队进行排列，把某个队排在它所击败的那个队之前。那么我们可以把整数1，2，…，n赋给这些队，以表示它们在这个排列中的序号。因为在比赛图中总是存在一条有向哈密顿路（定理5.12），所以这样的排序总是可能的，并且叫做用哈密顿路径来排序。

注意，用哈密顿路来排序与用得分来排序并不一定相同，因为比赛图可能有不止一条有向哈密顿路。在这种情况下，比赛图就可能有不止一种哈密顿路排序。然而，在传递比赛图中，恰好只有一条哈密顿路，下面的定理论述了这一点，而且是容易证明的。

定理5.15　在传递比赛图中，恰好只存在一条有向哈密顿路。　□

至于其它排序步骤，可参阅文献[5.6]，[5.7]和[5.8]。

5.9　进一步阅读

文献[5.5]和[5.9]是很好的参考书，它们给出了许多有关有向图的结论。文献[5.10]是一本致力于研究比赛图的专著，另外还可参阅文献[5.11]。

有向树可用于组合事物的计算机描述中。对于这个题目的详细讨论，可参阅文献[5.12]，[5.13]，[5.14]和[5.15]。

有向树还可用于其它方面的应用，比如象电网络的拓扑研究。在这类网络的拓扑研究中，人们感兴趣的是有向图的有向树的数目。这个问题将在第六章中讨论。有向树在网络理论中的应用将在第十三章中讨论。

文献[5.16]讨论了有向图的几种应用。

5.10　习　　题

5.1　G是无自环和无并行边的有向图。令$\max\{\delta^-, \delta^+\}=k$，证明：

（a）G有一条长度至少为k的有向路。

（b）G有一条长度至少为$k+1$的有向回路，如果$k>0$。

5.2　证明：若给无向图$G=(V, E)$的边标定方向，因此在有向图中，

$$|d^+(v)-d^-(v)|\leqslant 1,\ 对于所有v\in V$$

5.3　有向图的一个有向切割是一个切割$<S, \bar{S}>$，其中所有边的方向或者都指向S，或者都离开S。证明：有向图中的每条边或者属于一个有向割集，或者属于一个有向回路，但不会同时属于两者（这是称之为"弧着色引理"普遍结果的一种特殊情况，将在第十章中证明。见定理10.31）。

5.4　对于至少有一条边的有向图G，证明下列两种说法是等价的：

（a）G不存在有向回路。

（b）G的每一条边都在一个有向割集中。

5.5　对于至少有一条边的有向连通图G，证明下列说法是等价的：

（a）G是强连通的。

（b）G的每一条边都在一个有向回路中。

（c）G不存在有向割集。

5.6 证明：不含并行边和无自环的 n 个顶点有向图是强连通的，如果

$$\min\{\delta^-, \delta^+\} \geq \frac{n-1}{2}$$

5.7 证明：包含一个奇数长度回路的强连通图，也包含一个奇数长度的有向回路。

5.8 $G=(V, E)$是有向图，且

$(a)\ d^+(x)-d^-(x)=l=d^-(y)-d^+(y)$；

$(b)\ d^+(v)=d^-(v)$　　　　对于$V \in v-\{x, y\}$。

证明：G中含有 l 条边不相交的有向$x—y$路径。

5.9 证明：强连通图有一个有向生成通道。

5.10 由三个符号x、y和z构成的最长的循环序列是什么？在这个序列中，不重复出现四个符号的子序列。给出一个这样的序列。

5.11 找出一个含有七个 0 和七个 1 的循环序列，在这个序列中所有 4 位二进制序列（除 0000 和 1111 外）都是作为序列的子序列出现。

5.12 证明：$G_{s,n}$ 的一个有向哈密顿回路对应于 $G_{s,n-1}$ 的一个欧拉轨迹。$G_{s,n}$ 是否总有一个哈密顿回路？

5.13 证明：在有n个顶点、m（$m>2n$）条边的有向图中，有向欧拉轨迹的数目是偶数。

5.14 证明：无并行边和无自环的有向图$G=(V, E)$有一条有向哈密顿路，如果

$$d^+(V)+d^-(V) \geq n-1, \qquad \text{对所有}v \in V$$

5.15 证明：每个比赛图都是强连通的，或者是改变一条边的方向后刚好使原图变成强连通的比赛图。

5.16 如果 s 中没有两个顶点是邻接的话，子集$S \in V$就是图$G=(V, E)$的一个独立集，证明：无自环的有向图有一个独立集 S，使得通过一条长度不大于 2 的有向路，从 S 中的一个顶点到达在 G 中、而不在 S 中的顶点V，文献[5.17]。

这个结果的一个有趣推论是：一个比赛图包含这样一个顶点，它通过一条长度不大于 2 的有向路可以到达每一个顶点。

5.17 证明 n 个顶点比赛图的得分s_i满足

$$\sum_{i=1}^{n} s_i^2 = \sum_{i=1}^{n}(n-1-s_i)^2$$

5.11 参考文献

5.1 S.W. Golomb, *Shift Register Sequences*, Holden-Day, San Francisco, 1967.

5.2 M. Hall, Jr., *Combinatorial Theory*, Blaisdell, Walltham, Mass., 1967.

5.3 T. Van Aardenne-Ehrenfest and N.G. de Bruijn,"Circuits and Trees in Oriented Linear Graphs," *Simon Stevin*, Vol. 28, 203—217(1951).

5.4 J.W. Moon,"On Subtournaments of a Tournament,"*Canad. Math. Bull.*, Vol. 9, 297—301(1966).

5.5 C.Berge, *Graphs and Hypergraphs*, North Holland, Amsterdam, 1973.

5.6 J.A. Bondy and U.S.R. Murty, *Graph Theory with Applications*, Macmillan, London, 1976.

5.7 M.G. Kendall, "Further Contributions to the Theory of Paired Comparisons," *Biometrics*, Vol. 11,43—62(1955).

5.8 J. W. Moon and N. J. Pullman,"On Generalized Tournament Matrices," *SIAM*

Rev., Vol. 12, 381—399(1970).

5.9 F. Harary, *Graph Theory*, Addison-Wesley, Reading, Mass., 1969.

5.10 J.W. Moon, *Topics on Tournaments*, Holt, Rinehart and Winston, New York, 1968.

5.11 C. St. J. A. Nash-Williams, "Hamiltonian Circuits," in *Studies in Graph Theory*, Part II, MAA Press, 1975, pp. 301—360.

5.12 D. E. Knuth, *The Art of Computer Programming*, Vol. 1: *Fundamental Algorithms*, Addison-Wesley, Reading, Mass., 1968.

5.13 D.E. Knuth, *The Art of Computer Programming*, Vol. 3: *Sorting and Searching*, Addison-Wesley, Reading, Mass., 1973.

5.14 A. V. Aho, J.E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, Mass., 1974.

5.15 E.M. Reingold, J. Nievergelt, and N. Deo, *Combinatorial Algorithms: Theory and Practice*, Prentice-Hall, Englewood Cliffs, N.J., 1977

5.16 F. Harary, R.Z. Norman, and D. Cartwright, *Structural Models: An Introduction to the Theory of Directed Graphs*, Wiley, New York, 1965.

5.17 V. Chvátal and L. Lovász, "Every Directed Graph Has a Semi-Kernel," in *Hypergraph Seminar* (Eds. C. Berge and D. K. Ray-Chaudhuri), Springer, New York, 1974, p. 175.

第六章　图的矩阵

本章，我们将引入图的关联、回路、割集和邻接矩阵，并且建立有助于揭示一个图的结构的这些矩阵的若干性质。关联、回路和割集矩阵是在电网络的研究中产生的，这是因为这些矩阵是描述一个网络的基尔霍夫电压和电流方程的系数矩阵。这样，这些矩阵的性质及其所建立起来的有关结果将广泛地用于本书第二篇的讨论中。这里论述的邻接矩阵的性质构成了信号流图法的基础。信号流图法在线性系统的研究中是一个强有力的工具。本章6.11节将讨论信号流图的理论。

我们对关联、回路和割集矩阵的讨论，主要是相对于有向图的。但是，对于无向图，如果加法和乘法是模数域中的模2运算，即$GF(2)$，则这些讨论仍就是有效的。

6.1　关联矩阵

考虑一个具有n个顶点，m条边且无自环的图G。G的全顶点关联矩阵$A_c=[a_{ij}]$具有n行和m列，每行对应一个顶点，每列对应一条边。A_c的元素a_{ij}定义如下：

G是有向图时，

$$a_{ij}=\begin{cases}1, & \text{如果第 } j \text{ 条边与第 } i \text{ 个顶点相关联，且离开第 } i \text{ 个顶点}\\ -1, & \text{如果第 } j \text{ 条边与第 } i \text{ 个顶点相关联，且指向第 } i \text{ 个顶点}\\ 0, & \text{如果第 } j \text{ 条边与第 } i \text{ 个顶点无关联}\end{cases}$$

G是无向图时，

$$a_{ij}=\begin{cases}1, & \text{如果第 } j \text{ 边与第 } i \text{ 个顶点相关联}\\ 0, & \text{如果第 } j \text{ 边与第 } i \text{ 个顶点无关联}\end{cases}$$

A_c的一行称为G的一个关联矢量。图6.1(a)和(b)给出了两个图和它们的全顶点关联矩阵。

从上面的定义应该清楚地看出，A_c的每一列恰好含有两个非零元素，一个是$+1$，一个是-1。因此，我们可以从任意剩余的$n-1$行得出去掉的那一行。这样，A_c的任意$n-1$行都包含了关于A_c的全部信息，换句话说，A_c中的各行是线性相关的。

A_c的任意（$n-1$）行子矩阵A称为图G的一个关联矩阵。对应于A_c中某行但不在A中的顶点称为A的参考顶点。注意：

$$R(A)^{+}=R(A_c)\leqslant n-1 \tag{6.1}$$

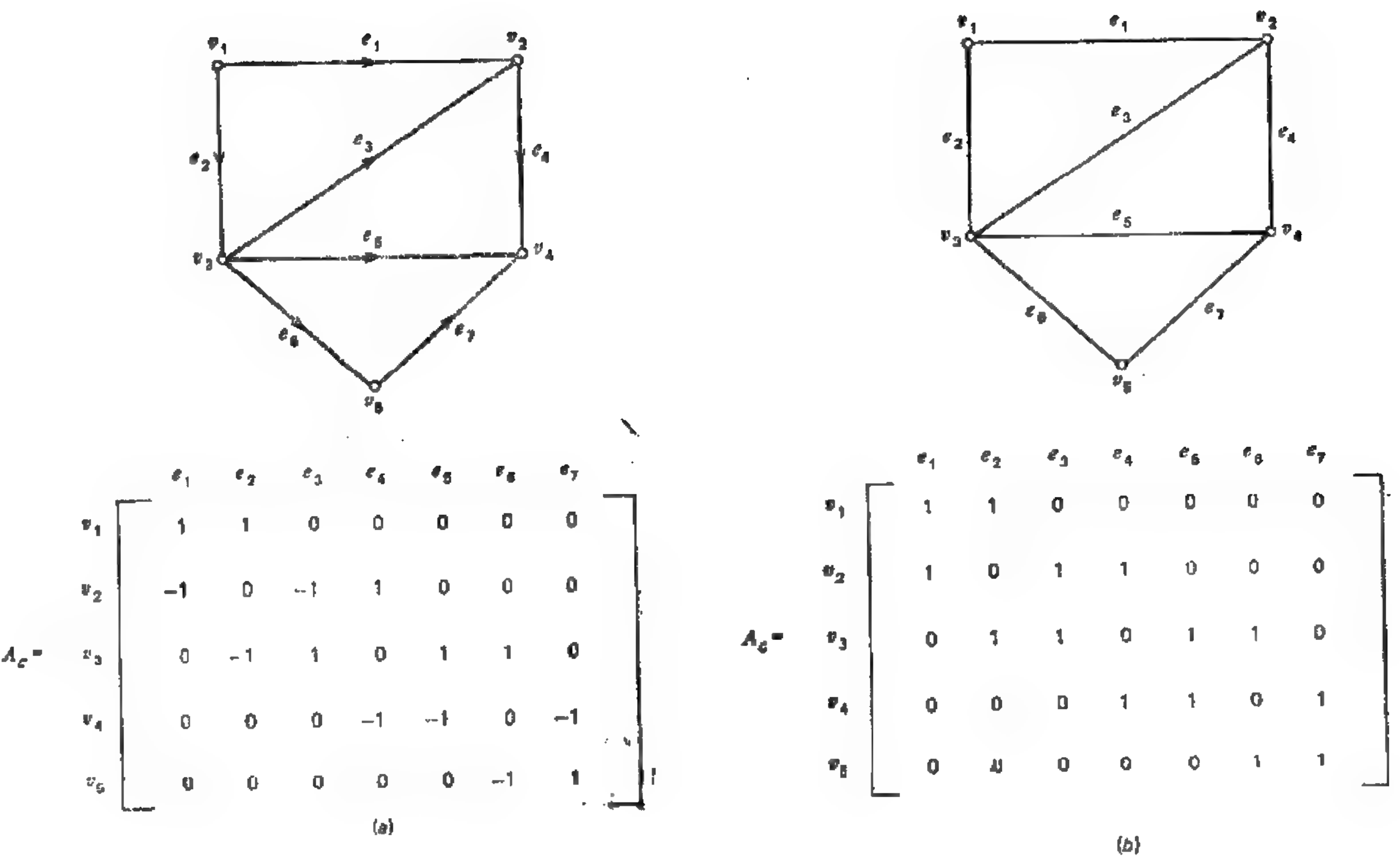
现在我们可以看出，对于连通图，A_c的秩实际上就等于$n-1$。这个结论基于下面的定理。

定理6.1　一个树的任意关联矩阵的行列式值等于±1。

证明

用一个树的顶点数n的归纳法来证明。

+　$R(A)$表示A的秩。——译者注



$$A_c = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array}\begin{array}{c} \begin{array}{ccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & -1 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix} \end{array}$$

$$A_c = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array}\begin{array}{c} \begin{array}{ccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \end{array}$$

图 6.1

(a)有向图G和它的全顶点关联矩阵；(b)无向图G和它的全顶点关联矩阵。

任何两个顶点的树的关联矩阵恰好是一个其值仅为±1的1×1矩阵。这样，对于$n=2$，定理成立。注意：对于$n=1$，定理不会产生。

设$2\leq n\leq k$时，定理成立。考虑具有$k+1$个顶点的一棵树T，令A示示T的一个关联矩阵。由定理2.5可知，T至少有两个悬挂顶点，令T的第i个顶点是一个悬挂顶点，但不是A的参考顶点。如果唯一与这个顶点相关联的边是第l条边，那么在A中

$$a_{il}=\pm 1\text{以及}a_{ij}=0,\ j\neq l$$

如果我们按第i行展开A的行列式，那么

$$\det(A)=\pm(-1)^{i+l}\det(A') \qquad (6.2)$$

这里，A'是由移去A中第i行第l列而得到的。

假设T'是从T中移去第i个顶点和第l条边后所得的图，显然，T'是一个树，这是因为T中第i个顶点是一个悬挂顶点，第l条边是一个悬挂边。并且容易检证，A'是T'的一个关联矩阵。由于T'是$n-1$个顶点的树，根据引入的假设，我们有

$$\det(A')=\pm 1 \qquad (6.3)$$

这个结果连同（6.2）式证明了，定理对于$n=k+1$是成立的。□

因为一个连通图至少有一个生成树，则由上面的定理可得如下结论：在n个顶点连通图的任何关联矩阵中，存在着一个$n-1$阶的非奇异子矩阵。这样，对于一个连通图，

$$R(A)=n-1 \qquad (6.4)$$

由于$R(A_c)=R(A)$，则得下面的定理。

定理6.2 n个顶点连通图G的全顶点关联矩阵的秩等于$n-1$，即图G的秩。□

上面定理的一个直接推论如下。

推论6.2.1 如果n个顶点的图具有P个连通片，那么它的全顶点关联矩阵的秩等于$n-p$，即图G的秩。□

6.2 切割矩阵

为了定义有向图的切割矩阵，我们必须对图的每个切割指定一个方向。

考虑有向图$G=(V,E)$。如果V_a是V的一个非空子集，那么回忆一下（第三章）使得，V_a中顶点与$\bar{V}_a$中顶点的相连接的边集就是一个切割，这切割记为$(V_a,\bar{V}_a)$。$(V_a,\bar{V}_a)$的方向可以假定为从V_a指向$\bar{V}_a$，或从$\bar{V}_a$指向V_a，我们假定规定的方向是从V_a指向$\bar{V}_a$。如果$(V_a,\bar{V}_a)$的一条边由V_a中的顶点指向$\bar{V}_a$中的顶点，那就说这条边的方向与切割$(V_a,\bar{V}_a)$的方向一致。

具有m条边的图G的切割矩阵$Q_c=[q_{ij}]$有m列和等于G中切割数的行。元素q_{ij}定义如下：

G是有向图

$$q_{ij}=\begin{cases}1, & \text{如果第}j\text{条边在第}i\text{个切割中，且方向与切割方向一致}\\ -1, & \text{如果第}j\text{条边在第}i\text{个切割中，且方向与切割方向相反}\\ 0, & \text{如果第}j\text{条边不在第}i\text{个切割中}\end{cases}$$

G是无向图

$$q_{ij}=\begin{cases}1, & \text{如果第} j \text{条边在第} i \text{个切割中}\\ 0, & \text{如果第} j \text{条边不在第} i \text{个切割中}\end{cases}$$

Q_c的每行称为一个切割矢量。

图6.1(a)中的有向图的三个切割如图6.2所示，每种情况的切割方向，如虚线箭头所示。对应于这三个切割的Q_c的子矩阵如下：

$$\begin{array}{c} \\ 切割1 \\ 切割2 \\ 切割3 \end{array}\begin{array}{c} \begin{array}{ccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 \end{array} \\ \left[\begin{array}{ccccccc} 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & -1 & 1 \\ 0 & -1 & 1 & 0 & 1 & 1 & 0 \end{array}\right]\end{array}$$

若将上面矩阵中的－1换成＋1，就可得到无向图情况所对应的子矩阵。

下面考虑任一顶点v，在相应的关联矢量中，非零元素代表着与v相关联的边，这些边形成了切割$\langle v,\ V-v\rangle$。如果我们假定这个切割的方向从v指向$V-v$，那么由切割和关联矩阵的定义可以看出，Q_c中对应于切割$\langle v,\ V-v\rangle$的行就等于A_c中对应顶点v的那一行，即A_c是Q_c的子矩阵。

现在我们开始证明Q_c的秩就于等A_c的秩，为此，我们需要有如下的定理。

定理6.3 切割矩阵Q_c中的每一行能够用两种方法表示成矩阵A_c的行的线性组合，在每种情况下，线性组合中的非零系数全为＋1，或者全为－1。

证明

设$\langle V_a, \overline{V}_a\rangle$是具有$n$个顶点、$m$条边的图$G$的第$i$个切割，$q_i$为相应的切割矢量，令$V_a=\{v_1,\ v_2,\ \cdots v_r\}$，$\overline{V}_a=\{v_{r+1},\ v_{r+2},\ \cdots v_n\}$。对于$1\leqslant i\leqslant n$，令$a_i$表示对应于顶点

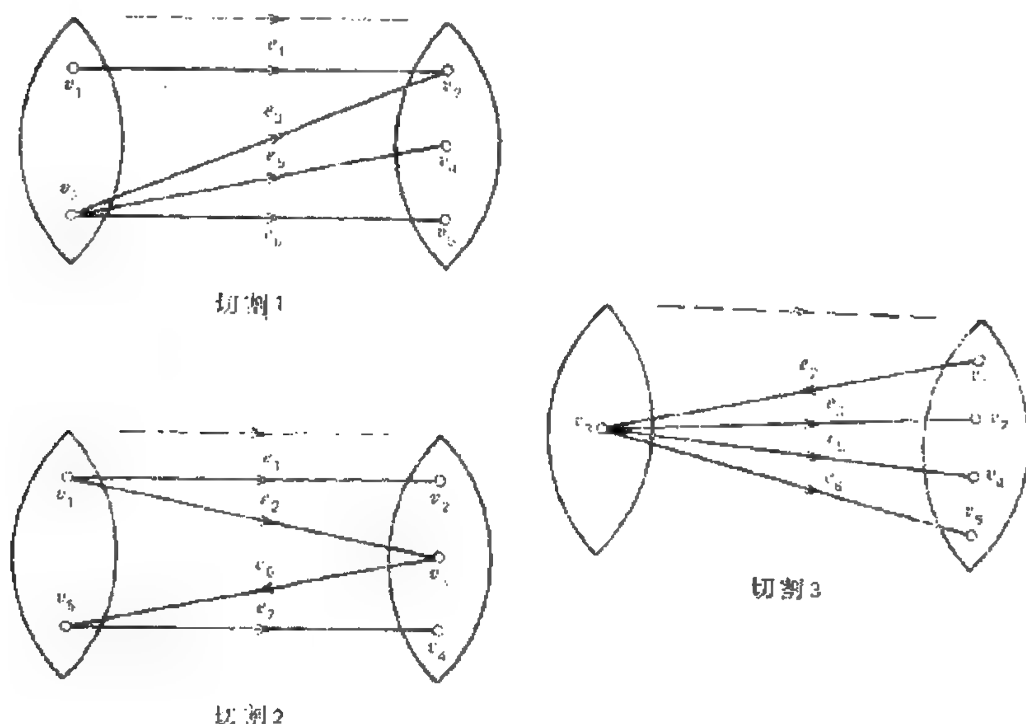


图 6.2 图6.1(a)中图的几个切割

v_i的关联矢量。

不失一般性，我们规定$\langle V_a, \overline{V}_a \rangle$的方向由$V_a$指向$\overline{V}_a$，并通过建立下式

$$q_i = a_1 + a_2 + \cdots + a_r = -(a_{r+1} + a_{r+2} + \cdots + a_n) \qquad (6.5)$$

来证明定理。

令v_p和v_q为第k条边的两个端点，$1 \leqslant k \leqslant m$，并设这条边由$v_p$指向$v_q$，则

$$a_{pk} = 1$$
$$a_{qk} = -1$$
$$a_{jk} = 0, \qquad j \neq p, \ q \qquad (6.6)$$

现在出现下面四种情况。

情况 1　$v_p \in V_a$和$v_q \in \overline{V}_a$，即$p \leqslant r$和$q \geqslant r+1$，所以$q_{ik} = 1$

情况 2　$v_p \in \overline{V}_a$和$v_q \in V_a$，即$p \geqslant r+1$和$q \leqslant r$，所以$q_{ik} = -1$

情况 3　v_p，$v_q \subset V_a$，即p，$q \leqslant r$，所以$q_{ik} = 0$

情况 4　v_p，$v_q \in \overline{V}_a$，即p，$q \geqslant r+1$，所以$q_{ik} = 0$

用(6.6)式很容易验证，在每种情况下，下面的等式都是成立的。

$$q_{ik} = (a_{1k} + a_{2k} + \cdots + a_{rk})$$
$$= -(a_{r+1,k} + a_{r+2,k} + \cdots + a_{nk}) \qquad (6.7)$$

由于上式对于所有的$1 \leqslant k \leqslant m$成立，从而得到(6.5)式。因此定理得证。 □

为了说明上面的定理，考虑图6.2中的切割1。这个切割使$V_a = \{v_1, v_3\}$中的顶点与$\overline{V}_a$中的顶点分离，切割方向由V_a指向$\overline{V}_a$。所以，相应的切割矢量就可以表示如下：

$$[1 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0] = a_1 + a_3$$
$$= -a_2 - a_4 - a_5$$

这里a_1，a_2，…，a_5是图6.1(a)中矩阵A的行。

定理6.3的一个重要的结论是$R(Q_c) \leqslant R(A_c)$。但是，由于A_c是Q_c的子矩阵，$R(Q_c) \geqslant R(A_c)$，因此，我们得到

$$R(Q_c) = R(A_c)$$

定理6.2和推论6.2.1分别引出下面的定理。

定理6.4　n个顶点连通图G的切割矩阵Q_c的秩等于$n-1$，即G的秩。 □

推论6.4.1　具有P个连通片、n个顶点的图G的切割矩阵Q_c的秩等于$n-P$，即G的秩。

正如上面的讨论所表明的，全顶点关联矩阵A_c是切割矩阵Q_c的一个重要的子矩阵。下面我们再给出Q_c的另一个重要的子矩阵。

我们知道，n个顶点连通图G的一个生成树确定了$n-1$个基本割集——一个基本割集对应于T中的一个树支。对应于这$n-1$个基本割集的Q_c的子矩阵就称为G的对于T的基本割集矩阵Q_f。

令b_1，b_2，…，b_{n-1}表示T的树支。假定我们按照下列规则安排Q_f的行和列：

1.　对于$1 \leqslant i \leqslant n-1$，第$i$列对应于树支$b_i$；

2. 第 i 行对应于由 b_i 所确定的基本割集。

此外，如果我们规定基本割集的方向选为与所确定的树支的方向一致，那么矩阵 Q_f 就具有如下的简便形式：

$$Q_f = [U \mid Q_{fc}] \quad (6.8)$$

这里 U 是 $n-1$ 阶的单位矩阵，它的列对应于 T 的树支。

例如，图6.1(a)的连通图中相对应于生成树 $T=\{e_1, e_2, e_6, e_7\}$ 的基本割集矩阵 Q_f 如下：

$$Q_f = \begin{array}{c} \\ e_1 \\ e_2 \\ e_6 \\ e_7 \end{array} \begin{array}{c} \begin{array}{cccc|ccc} e_1 & e_2 & e_6 & e_7 & e_3 & e_4 & e_5 \end{array} \\ \left[\begin{array}{cccc|ccc} 1 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right] \end{array} \quad (6.9)$$

由(6.8)式可以清楚地看出，Q_f 的秩等于 $n-1$，即 Q_c 的秩。这样，每个切割矢量（可能是集矢量）就可以表示为基本割集矢量的线性组合。

6.3 回路矩阵

一个回路可以沿顺时针方向或逆时针方向循环一周。我们所选择的循环的方向就定义为回路的方向。一个回路的方向可以画一个箭头来表示，如图6.3所示。

考虑一个边 e，v_i 和 v_j 是它的两个端点。假定这条边从 v_i 指向 v_j，其方向就是表示回路C的方向。当我们沿着回路C的方向循环时，如果 v_i 出现在 v_j 之前，那么我们说 e 的方向与回路的方向一致。例如，在图6.3中，e_1 的方向与回路方向一致，而 e_4 的方向与回路方向相反。

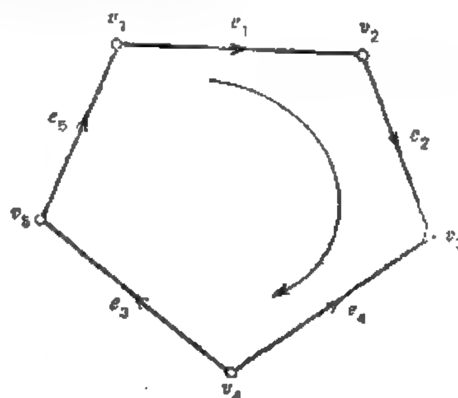


图 6.3

具有 m 条边的图G的回路矩阵 $B_c=[b_{ij}]$，有 m 列和等于图中回路数的行。元素 b_{ij} 定义如下：

图 G 是有向图

$$b_{ij} = \begin{cases} 1, & \text{如果第 } j \text{ 条边在第 } i \text{ 个回路中，且方向与回路方向一致} \\ -1, & \text{如果第 } j \text{ 条边在第 } i \text{ 个回路中，且方向与回路方向相反} \\ 0, & \text{如果第 } j \text{ 条边不在第 } i \text{ 个回路中} \end{cases}$$

图 G 是无向图

$$b_{ij} = \begin{cases} 1, & \text{如果第 } j \text{ 条边在第 } i \text{ 个回路中} \\ 0, & \text{如果第 } j \text{ 条边不在第 } i \text{ 个回路中} \end{cases}$$

B_c 的每行称为G的一个回路矢量。

举例，再次考虑图6.1(a)中的图。这个图的三个回路及其方向，如图

应于这三个回路的 B_c 的子矩阵如下：

$$
\begin{array}{r} \\ 回路1 \\ 回路2 \\ 回路3 \end{array}
\begin{array}{c}
\begin{array}{ccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 \end{array} \\
\left[\begin{array}{ccccccc}
1 & -1 & 0 & 1 & -1 & 0 & 0 \\
-1 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & -1 & 0 & 1 & 0 & -1 & -1
\end{array}\right]
\end{array}
$$

若将上面矩阵中的 -1 换成 +1，就可得到图6.1（b）中无向图所对应的子矩阵。

下面我们再给出B_e的一个重要的子矩阵。

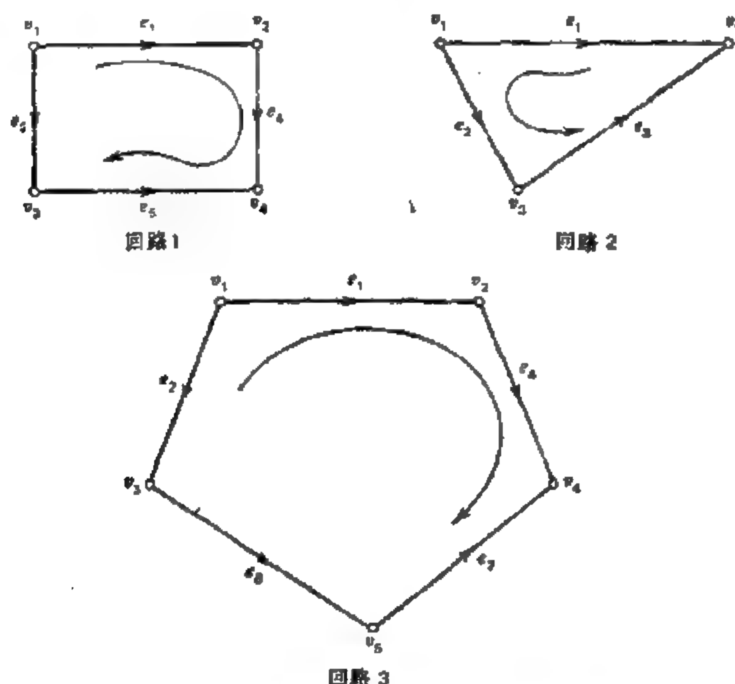


图 6.4　图6.1(a)所示图的几个回路

考虑具有n个顶点、m条边的连通图G的任一生成树。令 c_1，c_2，…，c_{m-n+1} 为T的弦。我们知道，这$m-n+1$个弦确定了$m-n+1$个基本回路的集合。对应于这些基本回路的B_e的子矩阵，称为G的关于生成树T的基本回路矩阵B_f。

假定我们按下面的规则安排B_f的列和行：

1. 对于 $1\leqslant i\leqslant m-n+1$，第$i$列对应于弦$c_i$；

2. 第i行对应于由c_i所确定的基本回路。

此外，如果我们选择基本回路的方向与确定的弦的方向一致，那么矩阵B_f就可写成

$$
B_f=[U \mid B_{ft}] \tag{6.10}
$$

这里，U是$m-n+1$阶的单位矩阵，它的列对应于T的弦。

例如，图6.1(a)中的图对应于生成树$T=\{e_1,\ e_2,\ e_6,\ e_7\}$的基本回路矩阵如下：

$$
B_f=\begin{array}{c} \\ e_3 \\ e_4 \\ e_5 \end{array}
\begin{array}{c}
\begin{array}{ccccccc} e_3 & e_4 & e_5 & e_1 & e_2 & e_6 & e_7 \end{array} \\
\left[\begin{array}{ccc|cccc}
1 & 0 & 0 & -1 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & -1 & -1 & -1 \\
0 & 0 & 1 & 0 & 0 & -1 & -1
\end{array}\right]
\end{array} \tag{6.11}
$$

由（6.10）式可以很明显地看出，B_f的秩等于$m-n+1$。由于B_f是B_c的子矩阵，则得

$$R(B_c)\geqslant m-n+1$$

在下一节我们将证明，在连通图的情况下，B_c的秩等于$m-n+1$。

注意，在上节中用来确定Q_c秩的方法不能用于B_c的情况（为什么？），这是否意谓着我们所看到的（第四章）回路和割集之间的“对偶性”仅仅是偶然的呢？不是的，我们将看到，下一节的论述会进一步证实这个“对偶性”。

6.4 正 交 关 系

在4.6节我们证明了，在无向图的情况下，每一个回路向量对于每个割向量是正交的。现在我们来证明，这一结论对于有向图的情况也是正确的。我们的证明基于下面的定理。

定理6.5 如果有向图中的一个切割和一个回路具有$2k$个公共边，那么其中的k条边在切割和回路中有相同的相关方向，剩余的k条边在切割中的方向与在回路中的方向相反。

证明

考虑有向图的切割$\langle V_a, \overline{V}_a\rangle$和回路$C$。假定我们从$V_a$中的一个顶点开始循环$C$。因此，对于每条引导我们从$V_a$的顶点走到$\overline{V}_a$的顶点的边$e_1$，总存在着另一条边$e_2$，将我们从$\overline{V}_a$的顶点引导到$V_a$中的顶点。定理的证明如下：如果$e_1(e_2)$在切割和回路中具有相同的相关方向，那么$e_2$（$e_1$）在切割中的方向与在回路中的方向相反（见图6.5）。 ▭

下面我们证明本节中的主要结论。

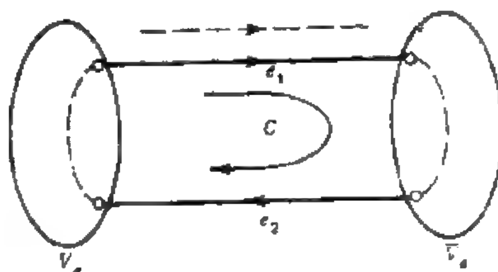


图 6.5

定理6.6（正交关系） 如果回路矩阵B_c和切割矩阵Q_c的列按相同的边序排列，那么

$$B_cQ_c^t=0$$

证明

考虑具有$2k$个公共边的一个回路和一个切割。相应的回路向量和切割向量的内积等于零，这是因为，由定理6.5可知，它们的内积是k个+1和k个-1之和。由于矩阵$B_cQ_c^t$的每个元素都是回路矢量和切割矢量的内积，所以定理得证。 ▭

正交关系是一个非常有意义的结论，它在图论和网络理论（如本书第二篇的讨论）中具有许多有趣的应用。现在我们用这个关系来确定回路矩阵B_c的秩。

考虑具有n个顶点、m条边的连通图G。设B_f和Q_f为G对于生成树T的基本回路矩阵和基本割集矩阵。如果B_f和Q_f的列按相同的边序排列，那么B_f和Q_f就可以写成如下形式：

$$B_f=[B_{ft}\quad U]$$

和

$$Q_f=[U\quad Q_{fc}]$$

由正交关系

$$B_fQ_f^t=0$$

$$[B_{ft} \;\; U]\begin{bmatrix} U \\ Q_{fe}^t \end{bmatrix} = 0$$

即
$$B_{ft} = -Q_{fe}^t \qquad (6.13)$$

令$\beta=[\beta_1,\ \beta_2,\ \cdots,\ \beta_\rho|\beta_{\rho+1},\ \cdots,\ \beta_m]$为一个回路矢量，它的列与$B_f$和$Q_f$的边序排列相同，这里$\rho$是$G$的秩。再次利用正交关系，则得

$$\beta Q_f^t=[\beta_1,\ \beta_2,\ \cdots,\ \beta_\rho|\beta_{\rho+1},\ \cdots,\ \beta_m]\begin{bmatrix} U \\ Q_{fe}^t \end{bmatrix} = 0$$

因此

$$[\beta_1,\ \beta_2,\ \cdots,\ \beta_\rho]=-[\beta_{\rho+1},\ \beta_{\rho+2},\ \cdots,\ \beta_m]Q_{fe}^t=[\beta_{\rho+1},\ \beta_{\rho+2}\cdots,\ \beta_m]B_{ft}$$

利用上式，我们可以将$[\beta_1,\ \beta_2,\ \cdots,\ \beta_m]$写成

$$[\beta_1,\ \beta_2,\ \cdots\beta_m]=[\beta_{\rho+1},\ \beta_{\rho+2},\ \cdots,\ \beta_m][B_{ft} \;\; U]=[\beta_{\rho+1},\ \beta_{\rho+2},\ \cdots,\ \beta_m]B_f \qquad (6.14)$$

这样，任何回路矢量都可以表示为基本回路矢量的线性组合。所以

$$R(B_c)\leqslant R(B_f)=m-n+1$$

将上不等式与（6.12）式联立，就可建立如下的定理和推论。

定理6.7 具有n个顶点、m条边的连通图G的回线矩阵B_c的秩等于$m-n+1$，即G的零度。 □

推论6.7.1 具有n个顶点、m条边、p个连通片的图G的回路矩阵B_c的秩等于$m-n+p$，即G的零度。 □

假定$\alpha=[\alpha_1,\ \alpha_2,\ \cdots,\ \alpha_\rho,\ \alpha_{\rho+1},\ \cdots,\ \alpha_m]$是一个切割矢量，它的列与$B_f$和$Q_f$的边序排列相同，因此从下面关系

$$\alpha B_f^t = 0$$

出发，用与建立（6.14）的相同步骤，就可证明

$$\alpha=[\alpha_1,\ \alpha_2,\ \cdots,\ \alpha_\rho]Q_f \qquad (6.15)$$

这样，每个切割向量就可以表示为一组基本割集矢量的线性组合。因为$R(Q_f)=n-1$，则有

$$R(Q_c)=R(Q_f)=n-1$$

上面就是定理6.4的另一种证明。

注意，两个子矩阵的环和相当于相应矢量的模2运算。因此，在无向图的情况下，（6.14）式和（6.15）式仅仅是我们在推导定理4.6的过程中所作证明的重新叙述，即一个回路（切割）可以表示成基本回路（基本割集）的环和。

6.5 切割、关联和回路矩阵的子矩阵

本节中，我们将描述对应于回路、割集、生成树和补生成树的Q_c、A_c和B_c的一些子矩阵，并且讨论这些子矩阵的一些性质。

定理6.8

1. 在切割矩阵Q_c的列之间存在着一种线性关系，这些列对应着一个回路中的一些

边。

2.在回路矩阵B_e的列之间存在着一种线性关系，这些列对应着一个割集中的一些边。

证明

1.我们把Q_e划分成列，使得

$$Q_e=[Q^{(1)},\ Q^{(2)},\ \cdots,\ Q^{(m)}]$$

令$\beta=[\beta_1,\ \beta_2,\ \cdots,\ \beta_m]$为一个回路矢量。由正交关系，则有

$$Q_e\beta^t=0$$

或者

$$\beta_1Q^{(1)}+\beta_2Q^{(2)}+\cdots+\beta_mQ^{(m)}=0 \tag{6, 16}$$

不失一般性，如果我们假定β的前r个元素不等于0，剩余的元素全为0，那么由(6.16)式得到

$$\beta_1Q^{(1)}+\beta_2Q^{(2)}+\cdots+\beta_rQ^{(r)}=0$$

这样，在Q_e的列$Q^{(1)}$，$Q^{(2)}$，…，$Q^{(r)}$之间存在着一种线性关系，这些列对应着一个回路中的一些边。

2.这种情形的证明用1中的相同步骤即可得到。 □

推论6.8.1 在关联矩阵的列之间存在着一种线性关系，这些列对应着一个回路中的一些边。

证明

因为此并联矩阵是Q_e的一个子矩阵，所以这个结论可由定理6.8的第一部分得出。 □

定理6.9 n个顶点连通图G的任意并联矩阵A的一个$n-1$阶子方阵是非奇异的，当且仅当对应于子矩阵的列的边能够形成G的一个生成树。

证明

必要性 考虑A的一个非奇异子矩阵的$n-1$列。因为这些列是线性无关的，所以由推论6.8.1得，在G的相应子图中不存在回路。又因为这个无回路的子图具有$n-1$条边，由定理2.2可知，它是图G的一个生成树。

充分性 证明可由定理6.1得出。 □

这样，连通图的生成树就与矩阵A的非奇异子矩阵一一对应了。这个结论事实上就是我们将在6.7节讨论的一个计算生成树公式的基础。

定理6.10 考虑具有n个顶点、m条边的连通图G。令Q是具有$n-1$行且秩为$n-1$的Q_e的子矩阵。Q的$n-1$阶子方阵是非奇异的，当且仅当对应于这个子矩阵的列的边能够形成G的一个生成树。

证明

必要性 设矩阵Q的列排列成

$$Q=[Q_{11}\quad Q_{12}]$$

Q_{11}是非奇异的。因为Q_{11}的各列线性无关，由定理6.8的第一部分得，在G的相应子图中不存在回路，这个无回路的子图具有$n-1$条边，因此由定理2.2得，它是G的一个生成树。

充分性 假定我们把Q的列排列成

$$Q=[Q_{11} \quad Q_{12}]$$

以及Q_{11}的列对应于生成树T的边。因此，对应于T的基本割集矩阵Q_f为

$$Q_f=[U \quad Q_{fc}]$$

因为Q的行可以表示为Q_f的行的线性组合，所以我们可以将Q写成

$$\begin{aligned} Q=[Q_{11} \quad Q_{12}]&=DQ_f \\ &=D[U \quad Q_{fc}] \end{aligned}$$

因此

$$Q_{11}=DU=D$$

而D是非奇异的，这是因为Q和Q_f都具有最大秩$n-1$。所以Q_{11}是非奇异的。定理的充分性得证。 □

下面介绍一个对偶的定理。

定理6.11 考虑具有n个顶点、m条边的连通图G。设B为具有$m-n+1$行、且秩为$m-n+1$的回路矩阵B_a的一个子矩阵。B的$m-n+1$阶子方阵是非奇异的，当且仅当这个子矩阵的列对应于一个补生成树的边。

证明

必要性 令矩阵B的列排列成

$$B=[B_{11} \quad B_{12}]$$

其中，B_{11}是非奇异的。因为B_{11}的各列线性无关，由定理6.8的第二部分得，对应的子图不包含G中的割集。而且这个子图具有$m-n+1$条边。这样，它就是图G的不含割集的最大子图（为什么？），因此也就是G的一个补生成树（见习题2.18）。

充分性 假定我们把B的列排列成

$$B=[B_{11} \quad B_{12}]$$

且B_{11}的列对应于G的补生成树的边。因此，对应于这个补生成树的基本回路矩阵是

$$B_f=[U \quad B_{fc}]$$

因为B的行可以表示成B_f的行的线性组合，所以我们可以把B写成

$$\begin{aligned} B=[B_{11} \quad B_{12}]&=DB_f \\ &=D[U \quad B_{fc}] \end{aligned}$$

因此$B_{11}=DU=D$

因为B和B_f都有最大秩$m-n+1$所以D是非奇异的，这样B_{11}就是非奇异的，定理的充分性得证。 □

为了说明定理6.10和6.11，考虑图6.1(a)中的图G以及（6.9）和（6.11）式所示的矩阵Q_f和B_f。对应于生成树$\{e_1, e_3, e_5, e_7\}$的Q_f的4阶子方阵和对应于补生成树$\{e_2, e_4, e_6\}$的B_f的3阶子方阵分别为

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

和

$$\begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & -1 \\ 0 & 0 & -1 \end{pmatrix}$$

可以验证这两个矩阵的解列式不为零，因此它们是非奇异的。

下面我们讨论一下并联矩阵的非奇异子矩阵的逆的一个有趣的性质，以结束本节。

定理6.12 设A_{11}是n个顶点连通图G的关联矩阵A的一个$n-1$阶非奇异子矩阵。因此，A_{11}^{-1}的每行中的非零元素或者全为1，或者全为-1。

证明

设A是以v_r为参考顶点的关联矩阵，假定

$$A=[A_{11}\quad A_{12}]$$

这里，A_{11}是非奇异的。由定理6.9得知，对应于A_{11}的列的边构成了G中的一个生成树。因此，对应于T的基本割集矩阵Q_f可以写成

$$Q_f=[U\quad Q_{fc}]$$

由定理6.3，每个切割向量可以表示为并联矩阵的行的线性组合，所以我们可以把Q_f写成

$$Q_f=[U\quad Q_{fc}]=D[A_{11}\quad A_{12}]$$

这样

$$D=A_{11}^{-1}$$

现在考虑Q_f的第i行q_i。令相应的割集为$\langle V_a,\ \overline{V}_a\rangle$。

设

$$V_a=\{v_1,\ v_2,\ \cdots,\ v_k\}$$

以及

$$\overline{V}_a=\{v_{k+1},\ v_{k+2},\ \cdots,\ v_n\}$$

假定割集$\langle V_a,\ \overline{V}_a\rangle$的方向由$V_a$指向$\overline{V}_a$。则由（6.5）式得到

$$q_i=a_1+a_2+\cdots+a_k \qquad (6.17)$$

$$=-(a_{k+1}+a_{k+2}+\cdots a_n) \qquad (6.18)$$

这里a_i是A_c的第i行。

注意，对应于v_r的行a_r不在A中。因此，如果$v_r\in V_a$，那么为了把q_i表示成A中行的线性组合，就必须把q_i写成（6.18）式的形式。如果$v_r\in\overline{V}_a$，则必须把q_i写成（6.17）式的形式。在这两种情况下，线性组合中的非零系数或者全为1，或者全为-1。

这样，$D=A_{11}^{-1}$的每一行中的非零元素或者全为1，或者全为-1。 □

上面定理的证明启发我们得出了下面计算A_{11}^{-1}的简单步骤。

1.由A_{11}构成了以v_n为参考顶点的关联矩阵A_{11}的树T。

2.将T的边标记为e_1，e_2，…，e_{n-1}，使得e_i对应于A_{11}的第i列。同样地将T的顶点标记为v_1，v_2，…，v_{n-1}，使得v_i对应于A_{11}的第i行。

3.令A_{11}^{-1}的第i列对应于v_i。对每个i，$1\leqslant i\leqslant n-1$，做如下操作：从$T$中移去$e_i$，并令$T_1$和$T_2$为产生的非连通图的两个片。设$V_a$为$T_1$的顶点集，$\overline{V}_a$为$T_2$的顶点集，并设$e_i$由

V_a中的顶点指向$\overline{V}_a$中的顶点。如果$v_i \in V_a$，那么，在A_{11}^{-1}的第i行中，对应于$\overline{V}_a$中顶点的所有列上的元素为－1；第i行中的其他元素均为0。如果$v_i \in \overline{V}_a$，那么在A_{11}^{-1}的第i行中，对应于V_a中顶点的所有列上的元素为1，第i行中的其他元素均为0。举例，令

$$A_{11}=\begin{matrix} & e_1 & e_2 & e_3 & e_4 \\ v_1 & 0 & -1 & 0 & 0 \\ v_2 & 1 & 1 & 1 & 0 \\ v_3 & -1 & 0 & 0 & 1 \\ v_4 & 0 & 0 & 0 & -1 \end{matrix}$$

关联矩阵A_{11}的树T，如图6.6(a)所示，参考顶点为v_5。

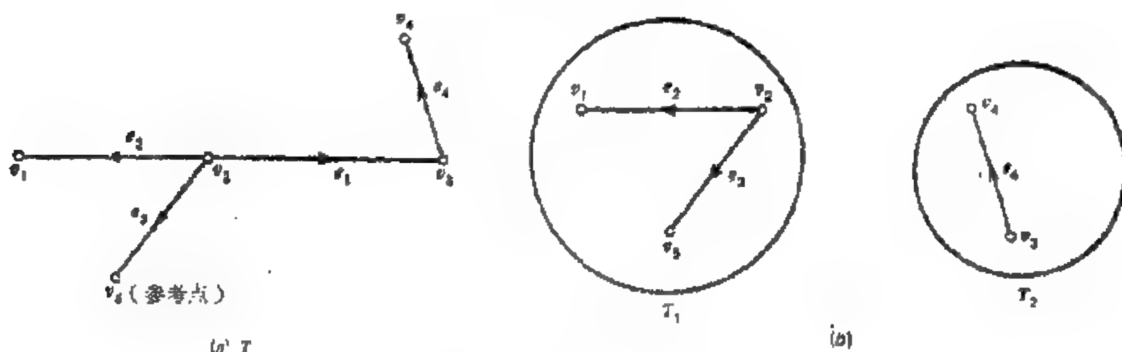


图 6.6 (a)树T；(b)子树T_1和T_2

为了得到A_{11}^{-1}的第一行，我们首先从T中移去e_1，从而得到两个片T_1和T_2，如图6.6(b)所示。这样

$$V_a=\{v_1,\ v_2,\ v_5\}$$
$$\overline{V}_a=\{v_3,\ v_4\}$$

因为参考顶点在V_a中，则在A_{11}^{-1}的第一行中，对应于$\overline{V}_a$中顶点v_3和v_4的列上的元素为－1，其结果如下：

$$[0 \quad 0 \quad -1 \quad -1]$$

以同样的方法，我们得到A_{11}^{-1}的所有其他行，A_{11}^{-1}如下：

$$A_{11}^{-1}=\begin{bmatrix} 0 & 0 & -1 & -1 \\ -1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

6.6 单位模矩阵

一个矩阵是单位模的，如果它的每个子方阵的行列式为1，－1或0。

本节中，我们将证明矩阵A_c，Q_f和B_f都是单位模的。

定理6.13 有向图的关联矩阵A_c是单位模的。

证明

我们对A_e的一个子方阵的阶进行归纳来证明定理。

显然，A_e的每个一阶子方阵的行列式为1，－1或0。作为归纳假设，假定每个阶数小于k的子方阵的行列式等于1，－1或0。

考虑A_e的任何k阶非奇异子方阵，这个矩阵的每列最多包含两个非零元素，一个为＋1，与（或）一个为－1。由于这个子矩阵是非奇异的，所以并非每列都有＋1和－1。同样的道理，在这个子矩阵中，没有一列是仅由0组成的。这样，至少有一列仅包含一个非零元素。将子矩阵的行列式按这一列展开，并用归纳假设，我们发现所期望的行列式为± 1。 □

令Q_f为n个顶点连通图G对于某个生成树T的基本割集矩阵。设T的树支为b_1，b_2，…，b_{n-1}。

令G'为将G的某个树支（如b_1）的两个端点短路后所得到的图。因此，$T-\{b_1\}$就是G'的生成树。现在让我们从Q_f中删去对应于树支b_1的行，并把所得矩阵记为Q'_f。不难证明：Q'_f是G'对于生成树$T-\{b_1\}$的基本割集矩阵，这样，从Q_f中删去任何行后所得的矩阵就是某个连通图的基本割集矩阵。．归纳上面，我们可以说，由Q_f的某些行形成的每个矩阵都是某个连通图的基本割集矩阵。

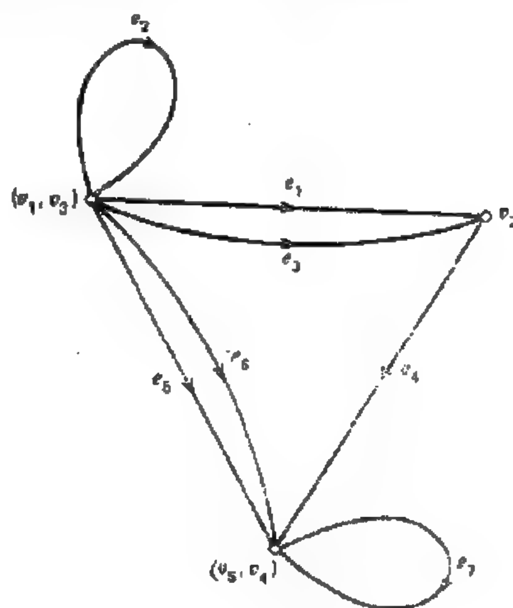


图 6.7

例如，考虑图6.1(a)中的图G，其基本割集矩阵Q_f如（6.9）式所示。由Q_f中对应于树支e_1，e_6的两行组成的子矩阵为

$$
\begin{array}{c|ccccccc|}
 & e_1 & e_2 & e_6 & e_7 & e_3 & e_4 & e_5 \\
e_1 & 1 & 0 & 0 & 0 & 1 & -1 & 0 \\
e_6 & 0 & 0 & 1 & 0 & 0 & 1 & 1
\end{array}
$$

可以验证，这个矩阵是图6.7所示图对于生成树$\{e_1, e_6\}$的基本割集矩阵，将图6.1(a)中的图的e_2，e_7两条边的顶点短接，就可得到这个图。

定理6.14 连通图G的任意基本割集矩阵Q_f都是单位模的。

证明

设Q_f是G对于生成树T的基本割集矩阵，因此

$$Q_f=[U \quad Q_{fc}]$$

将G的关联矩阵A划分为$A=[A_{11} \quad A_{12}]$，这里A_{11}的列对应于T的树支。由定理6.9得知，A_{11}是非奇异的。则我们可以把Q_f写成

$$Q_f=[U \quad Q_{fc}]=A_{11}^{-1}[A_{11} \quad A_{12}]$$

如果C是G的任意$n-1$阶子方阵，这里n为G的顶点数，D为A的相应的子矩阵，那么$C=A_{11}^{-1}D$。由于$\det D=\pm 1$或0，以及$\det A_{11}^{-1}=\pm 1$，因此得到

$$\det C = \pm 1 \text{ 或 } 0 \tag{6.19}$$

下面考虑Q_f的任意阶数小于$n-1$的子方阵H。从前面的关于本定理的讨论得知，H是某一连通图的基本割集矩阵的子矩阵。因此，$\det H = \pm 1$或0。下面的证明可以按证明（6.19）的方法进行。

这样，Q_f的每个子方阵的行列式为±1或0，因此Q_f是单位模的。 □

下面我们将证明B_f是单位模的。

定理6.15 连通图G的任意基本回路矩阵B_f是单位模的。

证明

令B_f和Q_f为G对于生成树T的基本回路矩阵和基本割集矩阵。如果$Q_f=[U \quad Q_{fc}]$，那么由（3.13）式得知

$$B_f = [-Q_{fc}^t \quad U]$$

由于Q_f是单位模的，所以Q_{fc}^t也是单位模的。要证明$[-Q_{fc}^t \quad U]$也是单位模的，就是一件简单的事了。 □

6.7 生成树的数目

本节，我们将导出一个计算连通图的生成树数目的公式。这个公式是在定理6.7的基础上，由一个称为比耐-柯西定理的矩阵理论导出的。

一个主行列式(或简称一个矩阵的大子式)是一个矩阵的最高阶行列式。令P是$p\times q$阶矩阵，Q是$q\times p$阶矩阵，且$p\leqslant q$。P和Q的主子式都是p阶的。如果p的一个大子式是由p中$i_1, i_2, \cdots, i_p$列组成，那么Q的相应大子式就是由Q中$i_1, i_2, \cdots, i_p$行所组成。例如，如果

$$P=\begin{bmatrix} 1 & -1 & 3 & 3 \\ 2 & 2 & -1 & 2 \end{bmatrix}$$

和

$$Q=\begin{bmatrix} 1 & 2 \\ 2 & -1 \\ -3 & 1 \\ 1 & 2 \end{bmatrix}$$

那么对于P的大子式

$$\begin{vmatrix} -1 & 3 \\ 2 & -1 \end{vmatrix}$$

Q的相应大子式为

$$\begin{vmatrix} 2 & -1 \\ -3 & 1 \end{vmatrix}$$

定理6.16（比耐-柯西定理） 如果P是$p\times q$矩阵，Q是$q\times p$矩阵，且$p\leqslant q$，那么

$\det(PQ)=\Sigma(P$和Q的相应大子式的乘积$)$ □

这个定理的证明见文献[6.1]。

举例，如果P和Q是如前给定的矩阵，那么由比耐-柯西定理可得

$$\det(PQ)=\begin{vmatrix}1 & -1\\ 2 & 2\end{vmatrix}\begin{vmatrix}1 & 2\\ 2 & -1\end{vmatrix}+\begin{vmatrix}1 & 3\\ 2 & -1\end{vmatrix}\begin{vmatrix}1 & 2\\ -3 & 1\end{vmatrix}+\begin{vmatrix}1 & 3\\ 2 & 2\end{vmatrix}\begin{vmatrix}1 & 2\\ 1 & 2\end{vmatrix}+\begin{vmatrix}-1 & 3\\ 2 & -1\end{vmatrix}\begin{vmatrix}2 & -1\\ -3 & 1\end{vmatrix}+\begin{vmatrix}-1 & 3\\ 2 & 2\end{vmatrix}\begin{vmatrix}2 & -1\\ 1 & 2\end{vmatrix}$$

$$+\begin{vmatrix}3 & 3\\ -1 & 2\end{vmatrix}\begin{vmatrix}-3 & 1\\ 1 & 2\end{vmatrix}$$

$$=-167$$

定理6.17 设G是连通无向图，A是由G的每条边任意标定方向后所得到的有向图的关联矩阵。因此

$$\tau(G)=\det(AA')$$

这里，$\tau(G)$是G的生成树的数目。

证明

由比耐-柯西定理

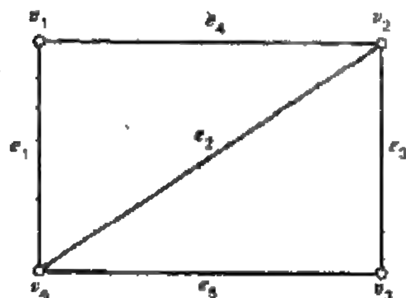
$$\det(AA')=\Sigma(A\text{和}A'\text{的相应大子式之积}) \tag{6.20}$$

注意：A和A'的相应大子式都具有等于1，-1或0的相同值（定理6.13）。因此（6.20）式右边求和式中的每个非零项的值为1。此外，A的大子式不为0，当且仅当对应于A的大子式的列的边形成一个生成树。

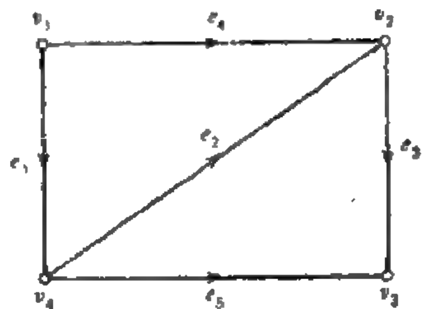
这样，（6.20）式右边求和式中的非零项与G的生成树之间就存在着一一对应的关系。因此，定理得证。 □

举例，考虑图6.8(a)所示图G。由图G的每条边标定任意的方向后所得的有向图，如图6.8(b)所示。此有向图的以顶点v_4为参考点的关联矩阵为

$$A=\begin{array}{c} \\ v_1\\ v_2\\ v_3\end{array}\begin{array}{c}\begin{array}{ccccc}e_1 & e_2 & e_3 & e_4 & e_5\end{array}\\ \begin{bmatrix}1 & 0 & 0 & 1 & 0\\ 0 & -1 & 1 & -1 & 0\\ 0 & 0 & -1 & 0 & -1\end{bmatrix}\end{array}$$



(a)



(b)

图 6.8

所以

$$AA^t=\begin{bmatrix}2 & -1 & 0\\ -1 & 3 & -1\\ 0 & -1 & 2\end{bmatrix}$$

和

$$\det(AA^t)=8$$

图G的 8 个生成树是

$\{e_1, e_4, e_5\}$, $\{e_1, e_3, e_4\}$, $\{e_1, e_3, e_5\}$, $\{e_1, e_2, e_3\}$,

$\{e_1, e_2, e_5\}$, $\{e_3, e_4, e_5\}$, $\{e_2, e_3, e_4\}$, $\{e_2, e_4, e_5\}$。

设v_1, $v_2\cdots$, v_n表示没有自环的无向图G的顶点。G的度矩阵$K=[k_{ij}]$是一个$n\times n$矩阵，其定义如下：

$$k_{ij}=\begin{cases}-p, & \text{如果}i\neq j\text{，且有}p\text{条联结顶点}v_i\text{和}v_j\text{的平行边}\\ d(v_i), & \text{如果}i=j\end{cases}$$

我们可以很容易看出，$K=A_cA_c^t$，并与我们在得到全顶点关联矩阵A_c时所选择边的方向无关。如果v_r是A的参考顶点，那么可通过移去K中的第r行第r列而获得AA^t。这样，定理6.17中的矩阵AA^t就可以直接由图G得到。

由度矩阵的定义可以清楚地看出，K中每行元素之和为零。同样，K中每列的元素之和也为零。具有这样性质的方阵称为等代数余子式矩阵。正如它的名字所暗示的那样，等代数余子式矩阵的所有代数余子式都相等。这样，由定理6.17就得到了下面的结论，它出自文献[6.3]。

定理6.18 一个连通的无向图的度矩阵的所有余子式都是相同的，且等于G的生成树的数目值。 □

下面推导一个公式用来计算在n个标记顶点上能造出的生成树的数目。显然这个数目和n个标记顶点完备图K_n的生成树的数目相同。

定理6.19（CAYLEY定理） 对于$n\geq 2$个顶点，存在着n^{n-2}个标记树。

证明

在K_n的情况下，矩阵AA^t具有如下形式

$$\begin{bmatrix}n-1 & -1 & \cdots & -1\\ -1 & n-1 & \cdots & -1\\ \vdots & \vdots & \vdots & \vdots\\ -1 & -1 & \cdots & n-1\end{bmatrix}$$

由定理6.17，这个矩阵的行列式给出了K_n的生成树的数目，这与n个顶点标记树数的目相同。

为了计算$\det(AA^t)$，要从AA^t中的其他各列减去AA^t的第一列。因此得到

$$\begin{bmatrix} n-1 & -n & -n & \cdots & -n \\ -1 & n & 0 & \cdots & 0 \\ -1 & 0 & n & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -1 & 0 & 0 & \cdots & n \end{bmatrix}$$

再将上面矩阵的其他各行加到第1行，使得到

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & n & 0 & \cdots & 0 \\ -1 & 0 & n & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -1 & 0 & 0 & \cdots & n \end{bmatrix}$$

这个矩阵行列式的值为n^{n-2}。因为一个矩阵的任意两行或两列相加，并不改变该矩阵行列式的值，于是定理得证。 □

Cayley定理[6.4]的几种证明在文献中都可见到，参见文献[6.5]和[6.6]

6.8 生成2-树的数目

本节，我们把图G的矩阵AA^t的代数余子式与适当形式的生成2-树的数目联系起来，为此，我们需要一些表示2-树的符号，在这个符号中，某些指定的顶点需要处于不同的连通片中。我们用符号$T_{ijk\cdots,pqr\cdots}$表示一个生成2-树，其中v_i，v_j，v_k，…在2-树的一个连通片中；v_p，v_q，v_r，…在另一个连通片中，图G中的这些生成2-树的数目用$\tau_{ijk\cdots,pqr\cdots}$表示。例如，在图6.8($a$)的图中，边$e_1$和$e_3$形成了一个$T_{14,23}$形式的生成2-树，且$\tau_{14,23}=1$。

下面，我们用A表示一个有向图的关联矩阵，这个有向图是通过把无向图G的每条边任意标记方向后而得到的。可是，我们把A称为G的关联矩阵。不失一般性，我们假定，v_n是A的参考顶点，A的第i行对应于顶点v_i，Δ_{ij}表示为AA^t的（i，j）余因式。

令A_{-i}表示从A中删去第i行后得到的矩阵。如果G'是将G中顶点v_i和v_n短接后所得到的图，那么可以很容易地验证下面两点：

1.A_{-i}是G'的关联矩阵，此时v_n作为参考顶点。

2.一个边的集合组成了G'的一个生成树，当且仅当这些边组成了G的一个生成2-树$T_{i,n}$。

这样，在A_{-i}的非零大子式与形如$T_{i,n}$的生成2-树之间存在着一一对应关系。

定理6.20 对于一个连通图G，

$$\Delta_{ii}=\tau_{i,n}$$

证明

显然，$\Delta_{ii}=\det(A_{-i}A_{-i}^t)$。因为$A_{-i}$的非零大子式对应于$G$的生成2-树$T_{i,n}$。反之亦然，所以定理立即得证。 □

下面考虑AA^t的（i，j）余子式Δ_{ij}，它由下式给出

$$\Delta_{ij}=(-1)^{i+j}\det(A_{-i}A_{-j}^t) \qquad (6.21)$$

由比耐-柯西定理，

$$\det(A_{-i}A^t_{-j}) = \Sigma(A_{-i}\text{和}A^t_{-j}\text{的相应大子式之积})。 \quad (6.22)$$

A_{-i}的每个非零大子式对应于一个形如$T_{ij,n}$的生成2-树，以及A_{-j}的每个非零大子式对应于一个形如$T_{ij,n}$的生成2-树。因此，（6.22）式右边求和式中的非零项对应于一个形如$T_{ij,n}$的生成2-树。这些非零项中的每一项都等于形如det（$F_{-i}F^t_{-j}$）的行列式，这里F是一个形如$T_{ij,n}$的生成2-树的关联矩阵。

定理6.21 令F表示一个以v_n为参考顶点的2-树$T_{ij,n}$的关联矩阵。如果F的第i行对应于顶点v_i，那么

$$\det(F_{-i}F^t_{-j}) = (-1)^{i+j}$$

证明

令T_1和T_2表示为$T_{ij,n}$的两个连通片。假定v_n在T_2中，通过相互交换矩阵FF^t的某些行和相应列，我们可以将矩阵AA^t写为

$$S = \left[\begin{array}{c|c} C & 0 \\ \hline 0 & D \end{array}\right]$$

这里

1. C是T_1的度矩阵；

2. D可以通过移去T_2的度矩阵中的对应于v_n的行和列而得到。

设S的第k'行对应于顶点v_k。

相互交换矩阵的某些行和相应的列并不能改变矩阵余子式的值。

因此

$$\begin{aligned} S\text{的}(i', j')\text{余子式} &= (FF^t)\text{的}(i, j)\text{余子式} \\ &= (-1)^{i+j}\det(F_{-i}F^t_{-j}) \end{aligned} \quad (6.23)$$

由定理6.18，C的所有余子式具有相同的值，且等于T_1的生成树的数目。因此我们有

$$C\text{的}(i', j')\text{余子式} = 1$$

而且

$$\det D = 1$$

所以

$$\begin{aligned} S\text{的}(i', j')\text{余子式} &= [C\text{的}(i', j')\text{余子式}] \\ &\quad [\det D] = 1 \end{aligned} \quad (6.24)$$

由（6.23）式和（6.24）式可得

$$\det(F_{-i}F^t_{-j}) = (-1)^{i+j}$$ □

上面定理的证明见文献[6.7]。

文献[6.14]得出了下面的结论。

定理6.22 对于连通图G，

$$\Delta_{ij} = \tau_{ij,n}$$

证明

因为（6.22）式右边求和式中的每个非零项都等于定理6.21中形式的行列式，我们得到

$$\det(A_{-i}\ A_{-j}^{t})=(-1)^{i+j}\tau_{ij,n}$$

所以

$$\begin{aligned}\Delta_{ij}&=(-1)^{i+j}\det(A_{-i}\ A_{-j}^{t})\\&=\tau_{ij,n}\end{aligned}$$

□

为了说明定理6.20和6.22，再次考虑图6.8(a)中的图。如果v_4作为A的参考顶点，那么

$$A_{-2}=\begin{bmatrix}1&0&0&1&0\\0&0&-1&0&-1\end{bmatrix}$$

及

$$A_{-3}=\begin{bmatrix}1&0&0&1&0\\0&-1&1&-1&0\end{bmatrix}$$

所以

$$\begin{aligned}\tau_{2,4}&=\det(A_{-2}\ A_{-2}^{t})\\&=\det\left(\begin{bmatrix}2&0\\0&2\end{bmatrix}\right)\\&=4\end{aligned}$$

以及

$$\begin{aligned}\tau_{23,4}&=(-1)^{2+3}\det(A_{-2}\ A_{-3}^{t})\\&=-\det\left(\begin{bmatrix}2&-1\\0&-1\end{bmatrix}\right)\\&=2\end{aligned}$$

在这个图中，形如$T_{2,4}$的生成2-树是

$$\{e_4,\ e_5\},\ \{e_3,\ e_4\},\ \{e_1,\ e_3\},\ \{e_1,\ e_5\}$$

而形如$T_{23,4}$的生成2-树是

$$\{e_1,\ e_3\},\ \{e_3,\ e_4\}.$$

6.9 有向图中有向生成树的数目

本节我们讨论一种方法，用它来计算一个具有特定顶点作为根的给定有向图中有向生成树的数目，这是由文献[6.8]提出的，实际上，这个方法是定理6.17给出的计算一个图生成树数目的推广，并且由下面所定义的入度矩阵给出。

一个无自环、且$V=\{v_1,\ v_2,\ \cdots,\ v_n\}$的有向图$G=(V,\ E)$的入度矩阵$K=[k_{pq}]$是一个$n\times n$阶矩阵，其定义如下：

$$k_{pq}=\begin{cases}-w, & \text{如果}p\neq q\text{，且有}w\text{条从}v_p\text{指向}v_q\text{的并行边}\\ d^-(v_p), & \text{如果}p=q\end{cases}$$

令K_{ij}表示从K中移去第i行、第j列后所得到的矩阵。

Tutte方法是建立在下面定理的基础上的。

定理6.23 无自环、且$V=\{v_1, v_2, \cdots, v_n\}$的有向图$G=(V, E)$是一个以$v_r$为根的有向树，当且仅当它的入度矩阵$K$具有以下性质：

1.
$$k_{pp}=\begin{cases}0, & \text{如果}p=r\\ 1, & \text{如果}p\neq r\end{cases}$$

2.
$$\det(K_{rr})=1$$

证明

必要性　假设给定有向图G是一个以v_r为根的有向树。很显然，G是无环的。因此，(见5.7节)仅当$i<j$时，采用(i, j)是G的有向边的形式用数字1，2，…，n来标记G的顶点。在这样的记数中，根顶节点的标号将为1。如果G的新入度矩阵K'的第i行、第i列对应于标号为i的顶点，那么我们可以很容易地看出，K'具有以下性质：

$$k'_{11}=0,$$
$$k'_{pp}=1, \quad \text{对于}p\neq 1$$
$$k'_{pq}=0, \quad \text{如果}p>q$$

所以

$$\det(K'_{11})=1$$

矩阵K'可以通过相互交换K的某些行和相应的列而得到。这样的变换不改变K的任何子矩阵的行列式的值。因此

$$\det(K_{rr})=\det(K'_{11})=1$$

充分性　假定图G的入度矩阵K满足定理所给定的两个性质。如果G不是一个有向树，则由性质1和定理5.4中论点5可知，它含有一个回路C。根顶点v_r不可能在C中，这样就意味着$d^-(v_r)>0$，或者对于C中的某些其他顶点v，$d^-(v)>1$，这与性质1相矛盾。用同样的方式，我们可以证明

1. C必是一个有向回路；
2. C中任一条边都与C中任意顶点相关联。

现在考虑K的一个子矩阵K_s，K_s是由对应于C中顶点的列所组成。由于上述性质，K_s中每个对应于C中一个顶点的行正好只有一个1和一个-1，而所有其他行只含有零元素。因此，K_s的列之和为零。换言之，K中对应于C中顶点的列之和为零。因为v_r不在C中，矩阵K_{rr}的情形也是正确的，这与性质2相矛盾。因此充分性得证。　□

下面推导Tutte的方法，此方法是计算有向图G的以v_r为根的有向生成树数目τ_d的，假定G无自环。

对于任意图g，令$K(g)$表示它的入度矩阵，K'表示在K中用零列替换第r列所得到的矩阵。S表示G的所有子图的集合，其中$d^-(v_r)=0$和$d^-(v_p)=1$（$p\neq r$）。显然

$$|S| = \prod_{p=1}^{n} d^-(v_p)$$

而且，对于任意子图$g\in S$，相应的入度矩阵满足定理6.23所给的性质1。

一个方阵的行列式是其列的线性函数，这是矩阵理论中的一个众所周知的结论。例如，如果

$$P=[p_1,\ p_2,\ \cdots,\ p_i'+p_i'',\ \cdots,\ p_n]$$

是一个列为p_1，p_2，…，$p_i'+p_i''$，…，p_n的方阵，那么

$$\det P=\det[p_1,p_2,\cdots,p_i',\cdots,p_n]+\det[p_1,p_2,\cdots,p_i'',\cdots,p_n]$$

利用行列式函数的线性性质以及矩阵$K'(G)$中每列元素之和为零的性质，我们就能把$\det K'(G)$写成满足定理6.23中性质1的每个行列式$|S|$之和。可以看出，在这些行列式与S的子图的入度矩阵之间存在着一一对应的关系。这样

$$\det K'(G)=\sum_{g\in S}\det K'(g)$$

所以

$$\det K'_{rr}(G)=\sum_{g\in S}\det K'_{rr}(g)$$

因为

$$\det K'_{rr}(G)=\det K_{rr}(G)$$

以及

$$\det K'_{rr}(g)=\det K_{rr}(g),\quad 对于所有g\in S$$

我们得到

$$\det K_{rr}(G)=\sum_{g\in S}\det K_{rr}(g)$$

由定理6.23可以得到如下结论：上式右边求和式中每一个行列式是非零的，且值为1，当且仅当S中相应的子图是一个有向生成树。这样，我们就已经证明了下面的定理。

定理6.24 设K为无自环有向图G的入度矩阵。令K的第i行对应于G的顶点v_i。因此，G的以v_r为根的有向生成树数目τ_d可由下式求出：

$$\tau_d=\det K_{rr}$$

这里，K_{rr}是移去K中的第r行第r列后所得到的矩阵。 □

现在我们举例说明上面的定理以及推导证明的论证过程。

考虑图6.9所示的有向图G，如试计算以v_1为根的有向生成树的数目。

G的入度矩阵K是

$$K=\begin{bmatrix} 1 & -1 & -2 \\ -1 & 2 & -1 \\ 0 & -1 & 3 \end{bmatrix}$$

以及

$$K'=\begin{bmatrix} 0 & -1 & -2 \\ 0 & 2 & -1 \\ 0 & -1 & 3 \end{bmatrix}$$

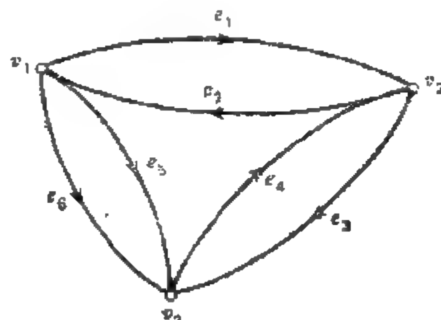


图 6.9

我们可以把K'写成

$$\det K'=\begin{bmatrix}0&-1&-2\\0&2&-1\\0&-1&3\end{bmatrix}=\begin{bmatrix}0&-1&-2\\0&1&-1\\0&0&3\end{bmatrix}$$

$$+\begin{bmatrix}0&0&-2\\0&1&-1\\0&-1&3\end{bmatrix}$$

$$=\begin{bmatrix}0&-1&-1\\0&1&0\\0&0&1\end{bmatrix}+\begin{bmatrix}0&-1&-1\\0&1&0\\0&0&1\end{bmatrix}$$

$$+\begin{bmatrix}0&-1&0\\0&1&-1\\0&0&1\end{bmatrix}$$

$$+\begin{bmatrix}0&0&-1\\0&1&0\\0&-1&1\end{bmatrix}+\begin{bmatrix}0&0&-1\\0&1&0\\0&-1&1\end{bmatrix}$$

$$+\begin{bmatrix}0&0&0\\0&1&-1\\0&-1&1\end{bmatrix}$$

上式右边的这六个行列式对应于如下边集的子图：

$$\{e_1,\ e_5\},\quad \{e_1,\ e_6\},\quad \{e_1,\ e_3\},$$
$$\{e_4,\ e_5\},\quad \{e_4,\ e_6\},\quad \{e_4,\ e_3\}$$

从上面的行列式中移去第一行第一列，我们得到

$$\det K_{11}=\begin{bmatrix}2&-1\\-1&3\end{bmatrix}=\begin{bmatrix}1&0\\0&1\end{bmatrix}+\begin{bmatrix}1&0\\0&1\end{bmatrix}+\begin{bmatrix}1&-1\\0&1\end{bmatrix}$$

$$+\begin{bmatrix}1&0\\-1&1\end{bmatrix}+\begin{bmatrix}1&0\\-1&1\end{bmatrix}+\begin{bmatrix}1&-1\\-1&1\end{bmatrix}$$

$$=5$$

以v_1为根的5个有向生成树是

$$\{e_1,\ e_5\},\ \{e_1,\ e_6\},\ \{e_1,\ e_3\},\ \{e_4,\ e_5\},\ \{e_4,\ e_6\}$$

6.10 邻接矩阵

设$G=(V,\ E)$为一个无平行边的有向图。令$V=\{v_1,\ v_2,\ \cdots,\ v_n\}$，G的邻接矩阵M

$=[m_{ij}]$是一个$n\times n$矩阵，其中m_{ij}定义如下：

$$m_{ij}=\begin{cases}1, & 如果(v_i, v_j)\in E;\\ 0, & 如果v_i到v_j间不存在边。\end{cases}$$

例如，图6.10中的图的邻接矩阵如下：

$$M=\begin{array}{c} \\ v_1\\ v_2\\ v_3\\ v_4\end{array}\begin{array}{c} \begin{array}{cccc}v_1 & v_2 & v_3 & v_4\end{array}\\ \begin{bmatrix}1 & 0 & 1 & 0\\ 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 1\\ 0 & 1 & 0 & 1\end{bmatrix}\end{array}$$

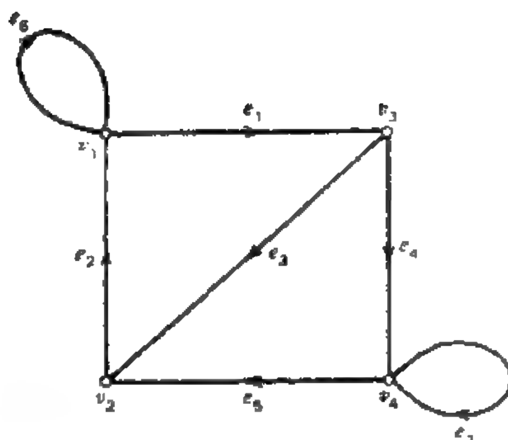


图 6.10

在无向图情况下，$m_{ij}=1$，当且仅当有一条连接v_i和v_j的边。

下面我们研究一些有关邻接矩阵的结论。

定理6.25 M^r的(i,j)元素$m_{ij}^{(r)}$等于从v_i到v_j且长度为r的有向通道数。

证明

用对r进行归纳来证明。很明显，结论对$r=1$成立。

作为归纳假设，我们假设定理对于M^{r-1}成立。因此

$$m_{ij}^{(r)}=\sum_{k=1}^{n}(m_{ik}^{(r-1)}\cdot m_{kj})$$

$$=\sum_{k=1}^{n}\begin{pmatrix}从v_i到v_k且长度为r-1的\\ 有向通道数\end{pmatrix}\cdot m_{kj}$$

对于每个k，上面求和式中相应的项给出了从v_i到v_j、且长度为r的有向通道数，其中最后一条边是从v_k到v_j。如果我们注意到，从v_i到v_j且长度为r的有向通道数$=\sum_{k=1}^{n}$(从v_i到v_j且长度为r的有向通道数，其中最后一边是从v_k到v_j)，则定理显而易证。 □

例如，考虑图6.10中图的矩阵M的三次方：

$$M^3=\begin{bmatrix}2 & 2 & 1 & 2\\ 1 & 1 & 1 & 1\\ 2 & 1 & 1 & 1\\ 2 & 1 & 1 & 1\end{bmatrix}$$

这个矩阵中的元素(1,4)给出了从v_1到v_4且长度为3的有向通道数，这些通道是

$$(e_6, e_1, e_4)和(e_1, e_4, e_7)$$

下面我们讨论由文献[6.9]提出的一个重要定理，它是形成求解线性代数方程的信号流图法的基础。为了引入这个定理，我们先给出一些术语。

有向图G的1-因子是G的一个生成子图，在这个子图中，每个顶点的入度和出度都等于1。显然，这样的子图是一个顶点-不连接的有向回路的集合，其中包含G的自环。例如，图6.10的两个1-因子，如图6.11(a)和(b)所示。

现在考虑整数1,2,…,n的一个排列j_1，j_2，…，j_n。如果重新排成1,2,3,…,n需要偶

数次交换，那么这个排列就是偶排列。用类似的方法，可以定义奇排列。例如，考虑排列 1 3 4 2。将它只要经过下面两次交换，就可重新排成 1 2 3 4：

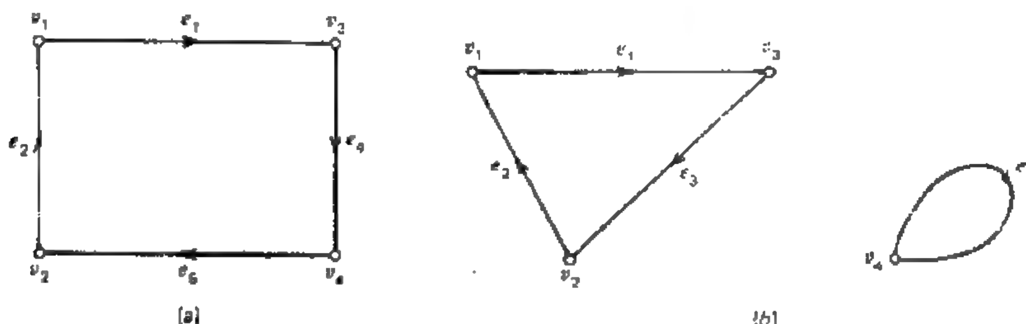


图 6.11　图6.10中图的两个1-因子

1. 交换 3 和 2；

2. 交换 3 和 4。

所以，这个排列是偶排列。

定理6.26　令H_i为一个n个顶点有向图G的1-因子，$i=1,2,\cdots,p$；L_i为H_i中的有向回路数；M为G的邻接矩阵。因此

$$\det M=(-1)^n\sum_{i=1}^{p}(-1)^{L_i}$$

证明

由行列式的定义得

$$\det M=\sum_{(j)}\varepsilon_{j_1j_2\cdots j_n}m_{1j_1}\cdot m_{2j_2}\cdots m_{nj_n}\qquad(6.25)$$

这里

1. $(j)=(j_1,j_2,\cdots,j_n)$是$1,2,\cdots,n$的一个排列；

2. 如果(j)是一个偶排列，那么$\varepsilon_{j_1j_2\cdots j_n}=1$，否则为$-1$；

3. 求和式$\sum_{(j)}$是对1，2，…，n的所有排列求和。

非零项$m_{1j_1}\cdot m_{2j_2}\cdots\cdots m_{nj_n}$对应于边集$(v_1, v_{j_1})$，$(v_2, v_{j_2})$，…$(v_n, v_{j_n})$。在这个集合中，每个$v_k$恰好出现两次，一次作为后一有序边的第一个元素，一次作为前一有序边的第二个元素。这就意谓着，在含有边(v_1, v_{j_1})，(v_2, v_{j_2})，…，(v_n, v_{j_n})的子图中，每个顶点的出度和入度都等于1。这样，在（6.25）求和式中的每个非零项对应于G的一个1-因子。相反地，每一个1-因子对应着一个非零项$m_{1j_1}\cdot m_{2j_2}\cdots\cdots m_{nj_n}$。

例如，图6.11(b)中的1-因子对应着项$m_{13}\cdot m_{32}\cdot m_{21}\cdot m_{44}$。

我们现在必须确定$\varepsilon_{j_1j_2\cdots j_n}$的符号。令$C$为对应于$j_1$，$j_2$，…，$j_n$的1-因子中的一个有向回路，且由下面$\omega$条边组成：

$$(v_{i_1}, v_{i_2}),\ (v_{i_2}, v_{i_3}),\ \cdots,\ (v_{i_\omega}, v_{i_1})$$

这些边的起始顶点形成了排列

$$i_1,\ i_2,\ \cdots,\ i_\omega$$

而它们的终止顶点形成了排列

$$i_2,\ i_3,\ \cdots,\ i_1$$

可以很容易地证明，只要经过$\omega-1$次交换，就可把i_2，i_3，…，i_1重新排列成i_1，i_2，…i_ω。

假设在对应于j_1，j_2，…，j_n的1-因子中存在着L个有向回路，令这些有向回路的长度为ω_1，ω_2，…，ω_L。因此我们需要经过

$$(\omega_1-1)+(\omega_2-1)+\cdots+(\omega_L-1)=W$$

次交换，就可把j_1，j_2，…，j_n重新排列成1，2，…，n。

所以
$$\varepsilon_{j_1,\ j_2,\cdots,\ j_n}=(-1)^W=(-1)^{n+L}$$

综上总结如下：

1. 每一个非零项$m_{1j_1}\cdots\cdots m_{nj_n}$对应于$G$的一个1-因子$H_i$（注意这一项的值为1）；

2. $\varepsilon_{j_1,\ j_2,\ \cdots,\ j_n}=(-1)^{n+L}$，这里$L_i$是$H_i$中有向回路的数目。

这样，（6.25）式就可化简为

$$\det M=(-1)^n\sum_{i=1}^{p}(-1)^{L_i}$$ □

例如，考虑图6.11中的1-因子，相应的L_i是

$$L_1=1$$
$$L_2=2$$

因此，图6.10中图的邻接矩阵行列式的值为

$$(-1)^1+(-1)^2=0$$

若直接展开行列式，即可证明这个结果。

假定在有向图G中，每条边$(i,\ j)$赋权w_{ij}。因此，邻接矩阵$M=[m_{ij}]$可定义如下：

$$m_{ij}=\begin{cases}w_{ij}, & \text{如果存在着一条从顶点}v_i\text{指向}v_j\text{的边}\\ 0, & \text{如果从顶点}v_i\text{到}v_j\text{间不存在边}\end{cases}$$

让我们定义G的一个子图H的权积$w(H)$为H的所有边的权之积。如果H没有边，即H是空图，则我们定义$w(H)=1$。这样，我们就能够得到定理6.26的一个简单的推广。

定理6.27 n个顶点、带权有向图G的邻接矩阵M的行列式为

$$\det M=(-1)^n\sum_H(-1)^{L_H}w(H)$$

这里H是G的一个1-因子，L_H是H中有向回路的数目。 □

6.11 考茨（Coates）图和梅森（Mason）图

本节，我们将推导出一个求解线性代数方程组的图论方法，即讨论分别由文献[6.10]和文献[6.11]、[6.12]提出的两个密切相关的方法。

6.11.1 考 茨 法

考虑由一组方程式所描述的线性系统

$$AX=Bx_{n+1} \tag{6.26}$$

这里$A=[a_{ij}]$是一个$n\times n$阶非奇异矩阵，X是未知变量x_1，x_2，…，x_n的列向量，B是元素b_1，b_2，…，b_n的列向量，x_n+1是输入变量。我们可以把x_k解出来

$$\frac{x_k}{x_{n+1}}=\frac{\sum_{i=1}^{n} b_i\Delta_{ik}}{\det A} \tag{6.27}$$

这里Δ_{ik}是A的（i，k）余因子式。

设A'表示由A得到的这样的矩阵，即将$-B$加到A的右边，然后再在所得矩阵的底下加上一零解所得的矩阵。现在我们将A'与带权有向图$G_c(A')$联系起来。

$G_c(A')$具有$n+1$个顶点x_1，x_2，…，x_{n+1}。如果$a_{ij}\neq 0$，那么在$G_c(A')$中存在着一条从x_i指向x_j、且权为a_{ij}的边。显然，A'是$G_c(A')$的邻接矩阵的转置。$G_c(A')$称为考茨流图，或简称与A'相关联的考茨图。有时把它也称为与（6.26）方程组相关联的考茨图。

举例，考虑方程组：

$$\begin{bmatrix} 2 & 1 & 1 \\ 1 & -1 & -2 \\ 2 & 1 & 2 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}=\begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}x_4 \tag{6.28}$$

这时的矩阵A'为

$$A'=\begin{bmatrix} 2 & 1 & 1 & -1 \\ 1 & -1 & 2 & 0 \\ 2 & 1 & 2 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

与（6.28）方程组相联系的考茨图$G_c(A')$，如图6.12（a）所示。

注意，我们可以把$G_c(A')$的每个顶点$x_i(1\leqslant i\leqslant n)$看作（6.26）式中相应的一个方程。例如，（6.26）式中的第i个方程可以这样得到，即所有指向x_i的边权和对应于顶点的变量乘积之和等于零。因而，与A相联系的考茨图$G_c(A)$，可以通过移去$G_c(A')$中的顶点v_{n+1}而得到。（6.28）式情况下的图$G_c(A)$，如图6.12（b）所示。

因为A是$G_c(A)$的邻接矩阵的转置，又因矩阵和它的转置矩阵的行列式具有相同的值，所以由定理6.27可得

$$\det A=(-1)^n\sum_H(-1)^{L_H}w(H) \tag{6.29}$$

这里H是在$G_c(A)$中的1-因子，$w(H)$是H的边权积，L_H是H中的有向回路数。这样，我们就能够以$G_c(A)$的1-因子的边权积形式，求出（6.27）中的分母。为了推导出与(6.27)式分子表达式的类似形式，我们需要求出Δ_{ik}。

$G_c(A)$中的从x_i指向x_j的1－因子连接H_{ij}是$G_c(A)$的一个生成子图，它含有(a)一条

从x_i指向x_j的有向路P；（b）一组包含了$G_c(A)$中除在P以外的所有顶点的顶点-不相接的有向回路。举例，图6.12(a)中的图$G_c(A')$的一个从x_4指向x_2的1-因子连接，如图6.12（c）所示。

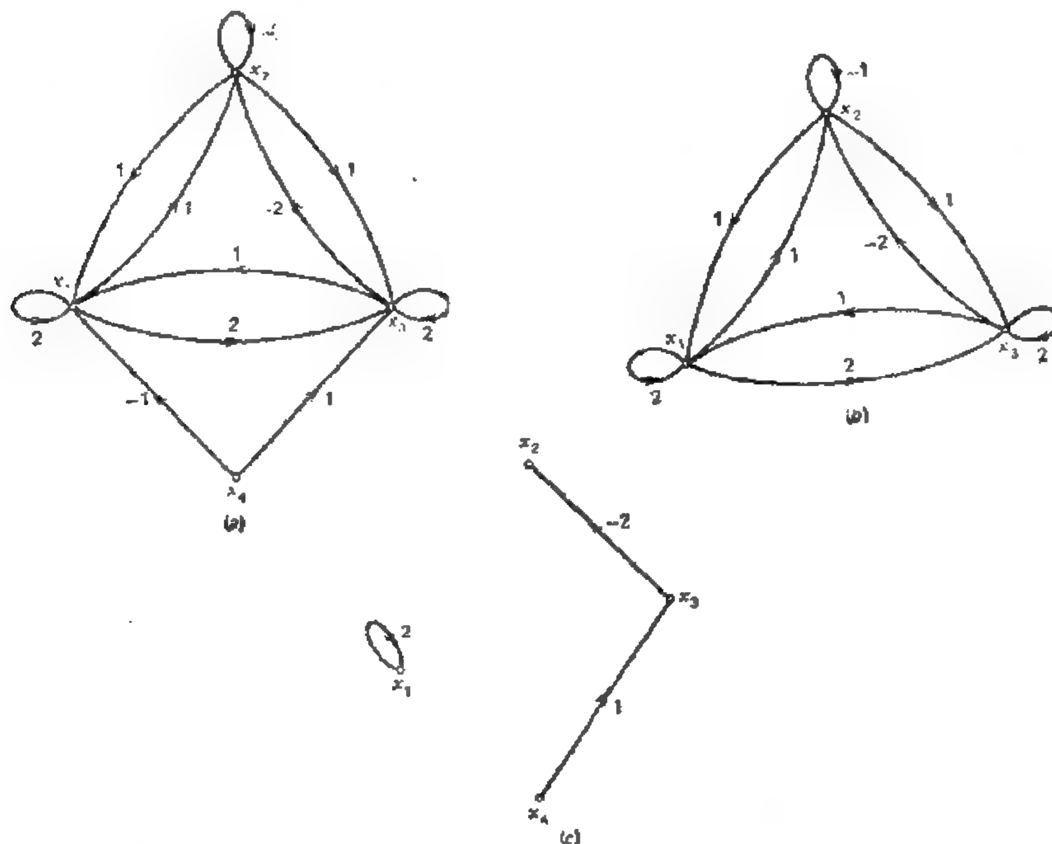


图 6.12 (a)考茨图$G_c(A')$；(b)图$G_c(A)$；(c)图$G_c(A')$连接H_{42}的1-因子。

定理6.28 设$G_c(A)$是一个与$n\times n$阶矩阵A相关联的考茨图。因此

1. $\Delta_{ii}=(-1)^{n-1}\sum_{H}(-1)^{L_H}w(H)$；

2. $\Delta_{ij}=(-1)^{n-1}\sum_{H_{ij}}(-1)^{L'_H}w(H_{ij})$，$i\neq j$；

这里

H是从$G_c(A)$中移去顶点x_i后所得图的1-因子，

H_{ij}是$G_c(A)$中的从顶点x_i指向x_j的1-因子连接，

L_H和L'_H分别是H和H_{ij}中的有向回路。

证明

1. 还可由定理6.27得出。

2. 注意到Δ_{ij}是这样的矩阵行列式，即除第i行元素为1以外，用一个零列代换A的第j列后所得到的矩阵行列式。令得到的矩阵用A_a表示。因此从$G_c(A)$中移去x_j的所有射出边（包括x_j处的自环），然后再加上一条从x_j指向x_i的权为1的边，就可得到考茨图$G_c(A_a)$。现在根据定理6.27，我们得到

$$\Delta_{ij}=\det A_a$$

$$=(-1)^{n}\sum_{H_\alpha}(-1)^{L_\alpha}w(H_\alpha) \qquad (6.30)$$

这里H_α是$G_c(A_\alpha)$中的1-因子，L_α是H_α中的有向回路数。

每个1-因子H_α必须含有数为1的添加边。若从H_α中移去这条边，我们就得到$G_c(A)$的一个1-因子连接H_{ij}，而且$w(H_\alpha)=w(H_{ij})$。我们也可以看出，在$G_c(A_\alpha)$中的H_α与$G_c(A)$中的H_{ij}之间存在着一一对应关系，即$w(H_\alpha)=w(H_{ij})$。因为H_{ij}中的有向回路数比H_α中的有向回路数少一个，所以我们有$L'_H=L_\alpha-1$。因此，如(6.30)式可得

$$\Delta_{ij}=(-1)^{n-1}\sum_{H_{ij}}(-1)^{L'_H}w(H_{ij}) \qquad \square$$

现在考虑$\sum_{i=1}^{n}b_i\Delta_{ik}$这一项，即(6.27)式的分子。它等于将$A$中的第$k$列用$B$代换后所得矩阵的行列式。我们很容易找出它与$A'_{n+1,k}$的关系（这里$A'_{n+1,k}$是移去$A'$中第$n+1$行第$k$列后所得矩阵），即

$$\sum_{i=1}^{n}b_i\Delta_{ik}=(-1)^{(n+1)-k-1}(-1)\det(A'_{n+1,k}) \qquad (6.31)$$

由定理6.28的第二部分可得

$$(-1)^{n+1+k}\det(A'_{n+1,k})=(-1)^{n}\sum_{H_{n+1,k}}(-1)^{L_H}w(H_{n+1,k}) \qquad (6.32)$$

这里L'_H是$G_c(A')$中的1-因子连接$H_{n+1,k}$的有向回路数。联立(6.31)式和(6.32)式，有

$$\sum_{i=1}^{n}b_i\Delta_{ik}=(-1)^{n}\sum_{H_{n+1,k}}(-1)^{L'_H}w(H_{n+1,k}) \qquad (6.33)$$

由(6.29)和(6.33)式，我们可得下面的定理。

定理 6.29 如果系数矩阵A是非奇异的，那么(6.26)式的解为

$$\frac{x_k}{x_{n+1}}=\frac{\sum_{H_{n+1,k}}(-1)^{L'_H}w(H_{n+1,k})}{\sum_{H}(-1)^{L_H}w(H)} \qquad (6.34)$$

其中$k=1$、2、…、n，这里

1. $H_{n+1,k}$是$G_c(A')$的从顶点x_{n+1}指向x_k的1-因子连接，

2. H是$G_c(A)$的1-因子，

3. L'_H和L_H分别是$H_{n+1,k}$和H的有向回路数。 $\square$

(6.34)式称为考茨增益公式。下面我们说明这个公式的一个应用，即解(6.28)式中的x_2/x_4。

与(6.28)式中矩阵A相关联的图$G_c(A)$（图6.12(b)）的1-因子，可由下面数积的形式给出。用括号将各有向回路中的顶点圈在一起，以区分1-因子中不同的有向回路。

1-因子	权积
$(x_1)(x_2)(x_3)$	-4
$(x_1)(x_2,\ x_3)$	-4
$(x_2)(x_1,\ x_3)$	-2
$(x_3)(x_1,\ x_2)$	2
$(x_1,\ x_2,\ x_3)$	1
$(x_1,\ x_3,\ x_2)$	-4

由上我们所得（6.34）式的分母为

$$\begin{aligned}\sum_H(-1)^{L_H}w(H)&=(-1)^3(-4)+(-1)^2(-4)+(-1)^2(-2)\\&\quad+(-1)^2(2)+(-1)(1)+(-1)(-4)\\&=4-4-2+2-1+4\\&=3\end{aligned}$$

$G_r(A')$（图6.12(a)）的从 x_4 指向 x_2 的 1-因子连接如下．有向路径中的顶点也用括号圈起来。

1-因子连接	权积
$(x_4,\ x_1,\ x_2)(x_3)$	-2
$(x_4,\ x_1,\ x_3,\ x_2)$	4
$(x_4,\ x_3,\ x_2)(x_1)$	-4
$(x_4,\ x_3,\ x_1,\ x_2)$	1

由上所得（6.34）式的分子为

$$\begin{aligned}\sum_{H'_{n+1,k}}(-1)^{L'_H}w(H_{n+1,k})&=(-1)(-2)+4+(-1)(-4)+1\\&=2+4+4+1\\&=11\end{aligned}$$

这样，我们得到

$$\frac{x_2}{x_4}=\frac{11}{3}$$

6.11.2 梅 森 法

为了推导梅森法，首先我们将（6.26）式重写为

$$x_j=(a_{jj}+1)x_j+\sum_{\substack{k=1\\k\neq j}}^{n}a_{jk}x_k-b_jx_{n+1},\qquad j=1,\ 2,\ \cdots,\ n$$

$$x_{n+1}=x_{n+1}$$

我们把上式写成矩阵形式：

$$(A'+U_{n+1})X'=X'$$

这里，A' 是前面已知定义的$(n+1)\times(n+1)$矩阵，U_{n+1}是 $n+1$ 阶单位矩阵，X' 是变量 $x_1,\ x_2,\ \cdots,\ x_{n+1}$的列向量。

考茨图 $G_c(A' : U_{n+1})$ 称为梅森信号流图，或简称为与矩阵 A' 关联的梅森图，记为 $G_m(A')$。例如，与方程(6.28)中所描述的系统相关联的梅森图 $G_m(A')$ 和 $G_m(A)$，如图6.13所示。

我们可以认为 $G_m(A')$ 中的每个顶点表示一个变量。如果存在一条从 x_i 指向 x_j 的有向边，那么我们就认为 x_i 对变量 x_j 产生了一个作用 $(a_{ji}x_i)$。因此，x_j 等于射入顶点 x_j 的边的数与对应于这些边的顶点变量乘积之和。这样，梅森图很简便直观地描述了一个系统中变量的流动。

注意，为了从给定的梅森图得到考茨图，我们仅需要从每个自环的数中减去1，以及对梅森图中每个无自环顶点添加一个数为－1的自环。这就等于说，考茨图 G_c 可以通过在梅森图 G_m 的每个顶点上单加一个数为－1的自环而得到。这样为了构造图 G_c，用 S 表示添加数为－1的自环集合。注意，这样构造的图 G_c，它的每个顶点处最多有两个，最少有一个自环。

现在考虑与矩阵 A 相关联的梅森图 $G_m(A)$，设 G_c 表示相应的考茨图。令 H 表示具有集合 S 中 j 个自环的 G_c 的1-因子，L_Q+j 表示 H 中有向回路的总数。若从 H 中移去集合 S 的 j 个添加自环，则得 G_m 的一个单一子图 Q，它是 L_Q 个顶点-不相接有向回路的组合，而且对于 $j<n$，

$$w(H)=(-1)^j w(Q) \qquad (6.35)$$

注意，如果 $j=n$，那么 Q 就是 G_m 的一个空子图，由定义可知它具有数 $w(Q)=1$。所以，在这种情况下，

$$w(H)=(-1)^n \qquad (6.36)$$

我们还可以看出，对于 G_m 的每个子图 Q 来讲，它都是 L_Q 个顶点不相接有向回路的组合，并且对应于 G_c 的一个单一1-因子，每个单一的1-因子由添加一个数为－1的自环（从 S 集中）到那些不在 Q 中顶点上而获得。

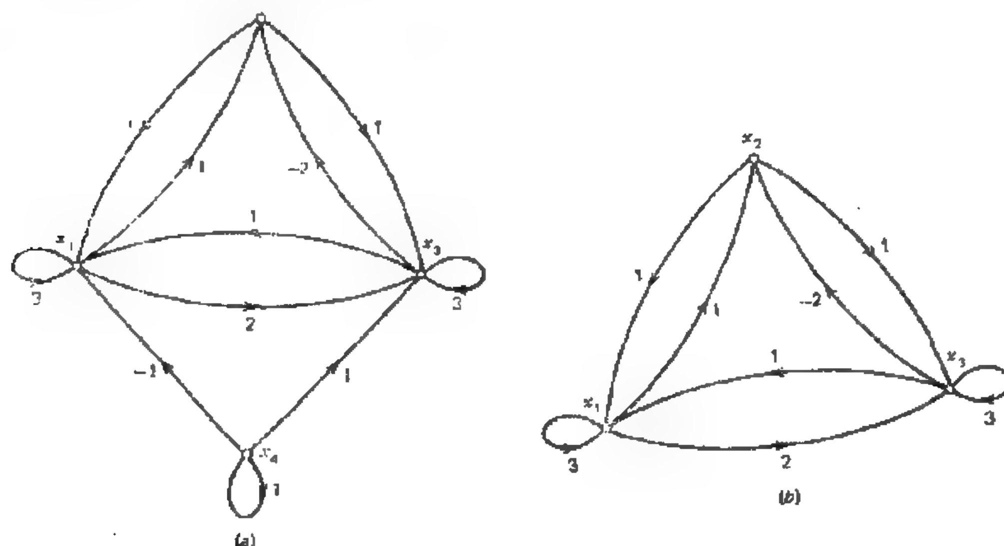


图 6.13 (a)梅森图 $G_m(A')$；(b)图 $G_m(A)$。

由定理6.27，则有

$$\det A=(-1)^n\sum_H(-1)^{L_Q+1}w(H)$$
$$=(-1)^n[1+\sum_Q(-1)^{L_Q}w(Q)] \qquad (6.37)$$

上式中的最后一步是由（6.35）式和（6.36）式得到的。

我们也可以把（6.37）式写成

$$\det A=(-1)^n[1-\sum_j Q_{j1}+\sum_j Q_{j2}-\sum_j Q_{j3}+\cdots] \qquad (6.38)$$

这里，Q_{jk}是G_m的第j个子图的权积，它是k个顶点-不相接的有向回路的组合。

这样，我们就可以把（6.27）式的分母表示成梅森图的适当的子图的数积形式。

我们把$(-1)^n\det A$称为图$G_m(A)$的行列式，记为Δ。因此利用（6.33）式和类似于推导（6.38）式的推理过程，我们就可以把（6.27）式的分子表示为

$$\sum_{i=1}^{n} b_i\Delta_{ik}=(-1)^n\sum_j w(P^j_{n+1,k})\Delta_j$$

这里，$P^j_{n+1,k}$是$G_m(A')$中从x_{n+1}指向x_k的第j条有向路径，Δ_j是$G_m(A')$的子图行列式，而这个子图与第j条有向路径$P^j_{n+1,k}$是顶点-不相接的。这样，我们就得到下面的定理。

定理 6.30 如果（6.26）式中的系数矩阵是非奇异的，则

$$\frac{x_k}{x_{n+1}}=\frac{\sum_j w(P^j_{n+1,k})\Delta_j}{\Delta}, \qquad k=1,\ 2,\ \cdots,\ n \qquad (6.39)$$

这里，$P^j_{n+1,k}$是$G_m(A')$中从x_{n+1}指向x_k的第j个有向路径，Δ_j是$G_m(A')$与第j条有向路径$P^j_{n+1,k}$顶点-不相接的子图行列式，Δ是图$G_m(A)$的行列式。□

（6.39）式称为梅森增益公式。在系统理论的文献中，$P^j_{n+1,k}$称为从顶点x_{n+1}指向x_k的正向路径。而且，把$G_m(A')$的有向回路称为反馈环路。

下面我们通过解（6.28）式的x_2/x_4来说明（6.39）式的应用。

$G_m(A)$（图6.13(b)）的顶点-不相接有向回路的不同组合和它们的数积为

有向回路	权积
(x_1)	3
(x_3)	3
$(x_1,\ x_2)$	1
$(x_2,\ x_3)$	−2
$(x_1,\ x_3)$	2
$(x_1,\ x_2,\ x_3)$	1
$(x_1,\ x_3,\ x_2)$	−4
$(x_1)(x_3)$	9
$(x_1)(x_2,\ x_3)$	−6
$(x_3)(x_1,\ x_2)$	3

由上所得（6.39）式的分母为

$$\Delta=1-(3+3+1-2+2+1-4)+(8-6+3)$$
$$=3$$

图$G_m(A')$中（图6.13(a)）从x_4指向x_2的不同有向路径和它们的权积为

j	$P^j_{4,2}$	权积
1	$(x_4,\ x_1,\ x_2)$	-1
2	$(x_4,\ x_1,\ x_3,\ x_2)$	4
3	$(x_4,\ x_3,\ x_2)$	-2
4	$(x_4,\ x_3,\ x_1,\ x_2)$	1

由$P^1_{4,2}$和$P^2_{4,2}$所得到的顶点-不相接有向回路分别为(x_3)和(x_1)。所以

$$\Delta_1=1-3=-2$$
$$\Delta_2=1-3=-2$$

由于$P^3_{4,2}$和$P^4_{4,2}$得到的顶点-不相接有向回路不存在，所以

$$\Delta_3=1$$
$$\Delta_4=1$$

这样，（6.39）式的分子为

$$P^1_{4,2}\Delta_1+P^2_{4,2}\Delta_2+P^3_{4,2}\Delta_3+P^4_{4,2}\Delta_4=2+4+4+1=11$$

所以

$$\frac{x_2}{x_4}=\frac{11}{3}$$

6.12 进一步阅读

本章的许多结论形成了本书第二篇中绝大部分内容的基础。文献[6.13]、[6.2]、[6.14]以及[6.15]都是与本章有关的一些参考文献。文献[6.2]还给出了现有结论在历史上的详细发展情况。

文献[6.16]着重介绍图论中枚举问题的研究，特别是那些与非标记图有关的问题。

计算生成树数目的问题在电网络理论的文献中已得到了足够的注意，计算一些特殊类型的生成树的递推关系是可行的，参看文献[6.17]、[6.18]、[6.19]、[6.20]以及[6.21]。文献[6.22]中包含了计算具有一定特性的生成树数目的公式。

文献[6.2]对于有关信号流图法的几个方面作了广泛的讨论。信号流图的几种应用，见文献[6.23]。关于信号流图在电网络中的应用，可以参看文献[6.24]。

6.13 习　　题

6.1 设$Q_f=[q_{ij}]$和$B_f=[b_{ij}]$分别为一个连通有向图的基本割集矩阵和基本回路矩阵，证明：

（a）如果（$q_{ij}q_{kj}$）和（$q_{ij}q_{kl}$）的乘积都不为0，那么它们一定相等。

（b）对B_f重做（a）。

6.2 G是一个具有m条边的连通有向图，设B是由G的回路矩阵中$\mu(G)$个独立行所形成的矩阵，证明：如果F是由元素1，－1和0组成的任意$\rho(G)\times m$阶矩阵，$\rho(G)$为其秩，并且满足$BF'=0$，那么F的每一行就是G的切割矩阵的每一行。

6.3 G是一个具有m条边的连通有向图，设Q是由G的切割矩阵中$\rho(G)$独立行所形成的矩阵。证明：如果F是由元素1，－1和0组成的任意$\mu(G)\times m$阶矩阵，$\mu(G)$为其秩，并且满足$QF'=0$，那么F的每行对应于G的一个回路或边-不相交接回路的并。

6.4 设$Q(x, y)$是一个连通且不可分的有向图G的切割矩阵Q_c的子矩阵，它仅仅含有Q_c中的那些对应于分隔顶点x和y的切割的行。证明：$Q(x, y)$包含了G的一个基本割集矩阵。（更一般性的结果，请参见文献[6.25]）

6.5 设B是平面图G的回路矩阵B_c的一个子矩阵，它仅仅含有B_c中的那些对应于G的网孔的行。证明：B是单位模的。

6.6 G是一个连通有向图，设B是由G的回路矩阵中任意$\mu(G)$个独立的行所形成的一个矩阵。证明：B的所有非奇异的大子矩阵的行列式具有相同的值。

6.7 证明定理4.11。

6.8 证明：对于连通有向图G，有

$$\tau(G)=\det(B_f\ B_f^t)=\det(Q_f\ Q_f^t)$$

这里B_f是G的基本回路矩阵，Q_f是G的基本割集矩阵。

6.9 G是一个连通有向图，设Q是由G的切割矩阵中任意$\rho(G)$个独立的行所形成的一个矩阵；B是由G的回路矩阵中任何$\mu(G)$个独立的行所形成的矩阵。证明：

$$\begin{bmatrix} Q \\ B \end{bmatrix}$$

是非奇异的。

6.10 证明：连通无向图G在域$GF(2)$上的回路和割集的子空间W_c和W_s是向量空间W_G的正交互补，当且仅当G的生成树的数目是奇数。

6.11 对于图G的任何边e，设$G\cdot e$表示从图中收缩边e后所得到的图。如果G是连通的，证明：

$$\tau(G)=\tau(G-e)+\tau(G\cdot e)$$

6.12 用习题6.11中的递推公式，求轮形图中的关于n个顶点的生成树的数目。（轮形图的定义见8.4节）

6.13 证明：如果e是K_n的一个边，那么

$$\tau(K_n-e)=(n-2)n^{n-3}$$

6.14 （a）设H是具有n个顶点的图，其中每对邻接顶点之间有k条平行边，G是H的基本简单图。证明：

$$\tau(H)=k^{n-1}\tau(G)。$$

（b）设H是将n个顶点连通图G中的每条边用一个长度为k的路径来替换后所得到的图。证明：

$$\tau(H)=k^{m-n+1}\tau(G)$$

这里m是G中边的数目。

6.15 确定定理6.17是定理6.24的一个特例。

6.16 一个每条边上都有权的图称为带权图。曾记得，G的一个子图的权积等于图的边的权之积，如若子图全为孤立顶点，则它的权积定义为1。

设A是以v_n为参考点的连通有向图G的关联矩阵。假定A的第i行对应于顶点v_i。令W是对应于G的权的对角矩阵，并假定A和W的列都以同样的边序排列。证明：

（a）$\det(AWA')=G$的所有生成树的权积之和。

（b）Δ_{ij}＝G的所有生成2-树的权积之和，这里Δ_{ij}是AWA'的（i，j）代表余子式。

6.17 一个无自环的有向图G的出度矩阵$K=[k_{ij}]$的定义如下：

$$k_{ij}=\begin{cases} -p, & \text{如果} i\neq j\text{，这里} p \text{是从} v_i \text{指向} v_j \text{的平行边数} \\ d^+(v_i), & \text{如果} i=j \end{cases}$$

再定义一个有向入树，它是一个具有顶点v_r，且其他每个顶点都有一条有向路径指向这个顶点的有向图，它的基础无向图是一个树。证明：树G的有向生成入树的数目可由类似于定理6.24的同样方法算出。

6.18 一个无自环的带权有向图的出度矩阵$K=[k_{ij}]$的定义如下：

（a）k_{ij}＝－（从v_i指向v_j的边权之和），$i\neq j$。

（b）k_{ii}＝从v_i出去的边权之和。

证明：$\det(k_{ii})$是G中以v_i为根的所有有向生成入树的权积之和。

注意 以v_i为根，意谓着从其他的每个顶点都有一条有向路径到v_i。

6.19 解下面的方程组：

$$\begin{bmatrix} 3 & -2 & 1 \\ -1 & 2 & 0 \\ 3 & -2 & 2 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}=\begin{bmatrix} 3 \\ 1 \\ -2 \end{bmatrix}x_4 \qquad (6.40)$$

6.20 设G_m是一个线性方程组的梅森图。证明：

（a）只要把指向顶点x_k的每条边的权都乘以因子$1/(1-a_{kk})$，就可移去x_k处的权$a_{kk}\neq 1$的自环。

（b）对于所有的$i\neq p$和$k\neq p$，将（a_{pi} a_{kp}）加到边（x_i，x_k）的权上，我们就可移去一个无自环的顶点x_p。

6.21 用习题6.20中的方法，将与（6.40）式相关联的梅森图化简成仅仅含有顶点x_4和x_1的一个图，并求出x_1/x_4。

6.14 参考文献

6.1 F.E. Hohn, *Elementary Matrix Algebra*, Macmillan, New York, 1958.

6.2 W.K. Chen, *Applied Graph Theory*, North Holland, Amsterdam, 1971.

6.3 G.Kirchhoff, "Über die Auflösung der Gleichungen, auf welche man bei der Untersuchung der linearen Verteilung galvanischer Ströme geführt wird," *Ann. Phys. Chem.*, Vol. 72, 497—508(1847).

6.4 A. Cayley, "A Theorem on Trees," *Quart J. Math.*, Vol. 23, 376-378(1889).

6.5 J.W. Moon, "Various Proofs of Cayley's Formula for Counting Trees," in *A Seminar on Graph Theory* (Ed. F.Harary and L. W. Beinke), Holt, Rinehart and Winston, New York, 1967, pp. 70—78.

6.6 H. Prüfer, "Neuer Beweis eines Satzes über Permutaionen," *Arch.Math.Phys.*, Vol. 27, 742—744(1918).

6.7 K. Sankara Rao, V.V. Bapeswara Rao, and V. G. K. Murti, "Two-Tree Admittance Products," *Electron. Lett.*, Vol. 6, 834—835(1970).

6.8 W. T. Tutte, "The Dissection of Equilateral Triangles into Equilateral Triangles," *Proc. Cambridge Phil. Soc.*, Vol. 44, 203—217(1948).

6.9 F. Harary, "The Determinant of the Adjacency Matrix of a Graph," *SIAM Rev.*, Vol.4, 202--210(1962).

6.10 C.L. Coates,"Flow-Graph Solutions of Lincar Algebraic Equations,"*IRE Trans. Circuit Theory*, Vol. CT-6, 170~187(1959).

6.11 S.J. Mason,"Feedback Theory: Some Properties of Signal Flow Graphs,"*Proc. IRE.*, Vol. 41, 1144—1156(1953).

6.12 S.J. Mason,"Feedback Theory: Further Properties of Signal Flow Graphs," *Proc. IRE.*, Vol. 44, 920—926(1956).

6.13 S. Seshu and M. B. Reed, *Linear Graphs and Electrical Networks,* Addison-Wesley, Reading, Mass., 1961.

6.14 W. Mayeda, *Graph Theory,* Wiley-Interscience, New York, 1972.

6.15 N. Deo, *Graph Theory with Applications to Engineering and Computer Science,* Prentice-Hall, Englewood Cliffs, N.J., 1974.

6.16 F. Harary and E. M. Palmer, *Graphical Enumeration,* Accdemic Press,New York, 1973.

6.17 B. R. Myers, "Numbdr of Trees in a Cascade of 2-Port Networks," *IEEE Trans. Circuit Theory,* Vol. CT-14, 284—290(1967).

6.18 B. R. Myers,"Number of Spanning Trees in a Wheel," *IEEE Trans. Circuit Theory,* Vol. CT-18, 280~282(1971).

6.19 S D. Bedrosian, "Number of Spanning Trees in Multigraph Wheels," *IEEE Trans. Circuit Theory,* Vol. CT-19, 77—78(1972).

6.20 N. K. Bose, R. Feick, and F K. Sun, "General Solution to the Spanning Tree Enumeration Problem in Multigraph Wheels," *IEEE Trans. Circuit Theory,* Vol. CT-20, 69—70(1973).

6.21 M. N. S. Swamy and K. Thulasiraman, "A Theorem in the Theory of Determinanis and the Number of Spanning Trees of a Graph," *Proc. IEEE Int. Symp. on Circuits and Systems,* 153—156(1976).

6.22 C. Berge, *Graphs and Hypergraphs,* North Holland, Amsterdam, 1973.

6.23 L. P. A. Robichaud, M. Boisvert, and J. Robert, *Signal Flow Graphs and Applications,* Prentice-Hall, Englewood Cliffs, N. J.,1962.

6.24 N. Balabanian and T. A. Bickart, *Electrical Network Theory,* Wiley, New York, 1969.

6.25 Y. Kajitani and S. Ueno, "On the Rank of Certain Classes of Cutsets and Tie-Sets of a Graph," *IEEE Trans. Circuits and Sys.*, Vol CAS-26, 666—668(1979).

第七章　平面性和对偶性

本章将讨论图论中的两个重要概念，即平面性和对偶性。我们首先考虑平面图，并推导平面图的一些性质。我们还要讨论由Kuratowski、Wagner、Harary、Tutte和Maclane提出的平面图特性。然后，我们讨论由Whitney给对偶图下的定义，它是通过回路和割集给出的，并把这个概念和看来好象无关的平面性概念联系起来。

在电网络理论中，对偶性是很有趣的，这是由于电网络中的电压和电流是对偶变量。这些变量的对偶性表现在基尔霍夫定律中。基尔霍夫电压定律是通过回路来建立的，而基尔霍夫电流定律是通过割集来建立的。在第二章和第四章中我们已经知道，回路和割集之间是具有对偶性的。在第十章中这种对偶性将变得明显，到时我们将证明：一个图的回路集合和割集集合具有相同的代数结构。

7.1　平　面　图

如果图G能画在S面上，并使它的边仅在端点处相交，那么就说它可嵌入S面。如果一个图能嵌入在一个平面上，则称它是可平面的。平面图G的这种画法称为G的可平面嵌入。

一个图的两种可平面嵌入，如图7.1所示。在其中一种可平面嵌入（图7.1（a））中，所有边都是直线，而在另一种可平面嵌入（图7.1（b））中，有一条边画成了曲线。注意，在图7.1（b）中，如果所有直线边都如图所画，那么联接顶点a和d的边，就不能画成直线。

很显然，如果一个图有自环或平行边，那么就没有一个可平面嵌入能使它的所有边画成直线。这会很自然地提出这样一个问题，是否每一个简单平面图G中都存在一个可平面

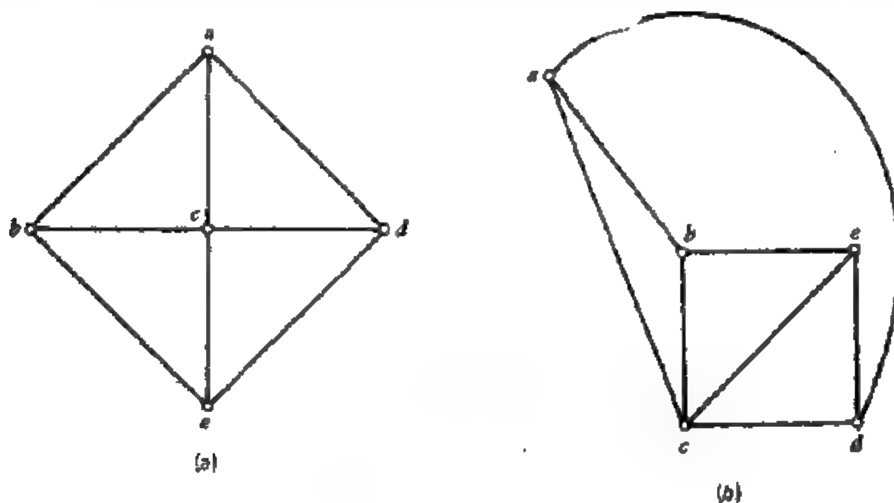


图 7.1　一个图的两种可平面嵌入

嵌入，从而使G的所有边都能画成直线呢？这个问题的回答是肯定的，下面的定理给予说明。

定理 7.1 对于每一个简单平面图，都存在一个可平面嵌入，从而使它的所有边都能用直线画出。 □

这个结论已由文献[7.1]、[7.2]和[7.3]分别证明过。

如果一个图不能嵌入一个平面，那么它就可能嵌入一些其它面。但是，我们将表明在平面上的可嵌入性和在球面上的可嵌入性是等价的；也就是说，如果一个图可嵌入一个平面，那么它必定能嵌入一个球面，反之亦然。这个结论的证明，要用到所谓平面上球的测地投影。下面就描述平面上球的测地投影。

假设我们放一个球在平面上（图7.2），并把与平面接触的点称为南极，与球上直径方向相反的点称为北极N。设P为球上任一点，那么由P到N的连线延长到平面上点P'，则称P'为P点在平面上的测地投影。显而易见，在球面上的点和它们在平面上的测地投影是一一对应的。

定理 7.2 一个图G可嵌入一个平面，当且仅当它能嵌入在一个球面上。

证明 设G'是G在球面上的一个嵌入。把球放到平面上，并使得北极既不是G'的顶点，也不是G'边上的点。

于是，在测地投影下，因为G'的边仅在端点相交，而且球上的点与其测地投影图象上的点是一一对应的，所以G'的映象是在平面上G的一个嵌入，

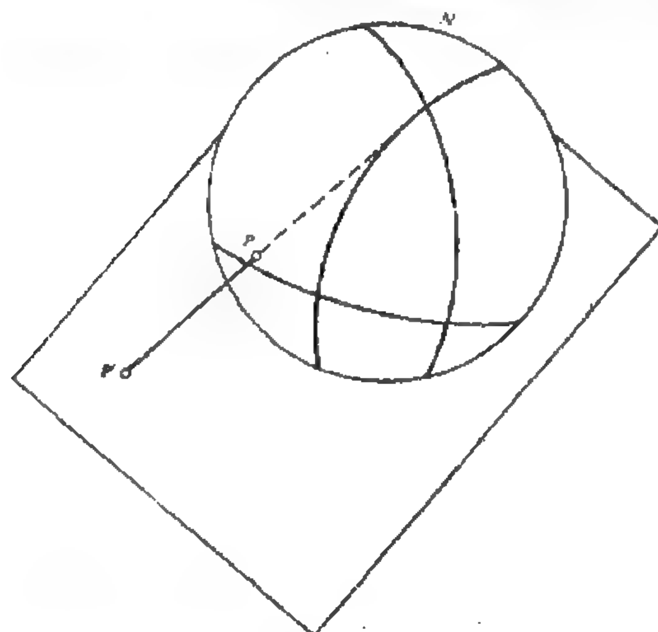


图 7.2 测地投影

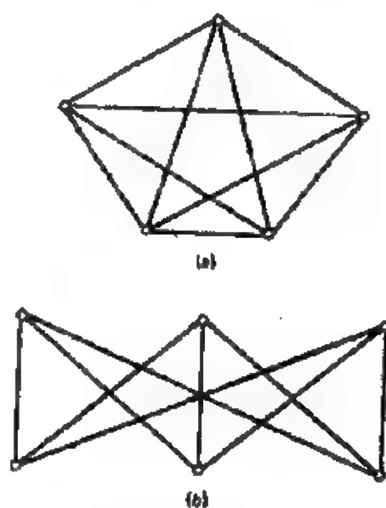


图 7.3 基本非平面图
(a)K_5; (b)$K_{3,3}$

同理可以证明相反的情况。 □

二个称为Kuratowski图的基本非平面图，如图7.3所示。其中之一是K_5，它是具有5个顶点的完备图；另一个是$K_{3,3}$。因为Kuratowski指出[7.3节]这两个图在表征平面性重要特征方面起着基础的作用，所以称为基本非平面图。这两个图的非平面性将在下节中

证实。

在结束本节之前，我们应当指出，文献[7.4]已证明了一个可分图是可平面的，当且仅当它的每个块都是可平面的。所以当考虑嵌入平面的问题时，我们只能考虑不可分图的嵌入性。

7.2 欧拉公式

当一个平面图嵌入在平面上时，便把这个平面分成若干个域。如果域所包括的面积有限，则称为有限域。否则，称为无限域。

例如，在图7.4所示的平面图中，斜线所示域f_5为无限域，f_1、f_2、f_3和f_4都是有限域。

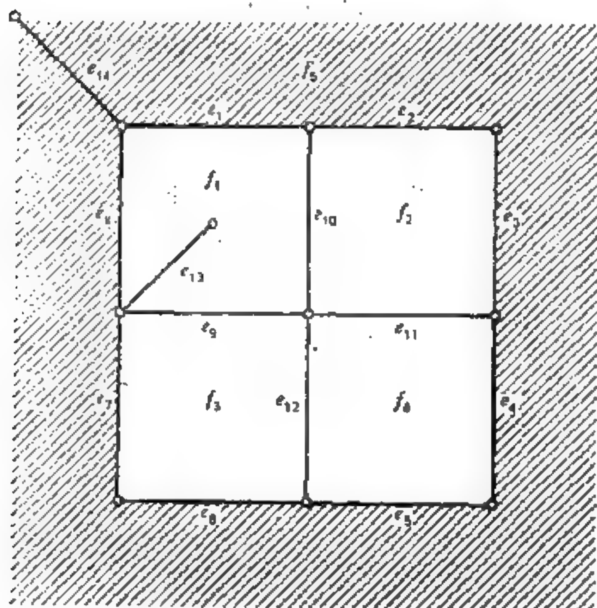


图 7.4

很显然，在域分界上的边恰好包含一个回路，并说这个回路包围了这个域。例如，边e_1、e_8、e_9、e_{10}和e_{13}形成了域f_1，如图 7.4 所示。他们包含了回路$\{e_1, e_8, e_9, e_{10}\}$。

注意，在任何一个平面图的球嵌入中，每一个域都是有限的。

假设我们把一个平面图嵌入在一个球面上，并把该球放置于一个平面上，以使球的北极在任一给定域的内部，譬如，域f。那么f的测地投影下的图象将是一个无限域。因此一个平面图总可以嵌入在一个平面上，以使任一选定域变为无限域。

设f_1，f_2，……f_r为平面图的域，f_r为无限域。我们用C_i表示域f_i边界上的回路，$1 \leq i \leq r$。

对应于各有限域的回路C_1，C_2，……C_{r-1}称为网孔。

容易证明，任何$k \geq 2$个网孔，譬如，包围了域f_1，f_2，…，f_k的C_1，C_2，…，C_k，这些网孔的环和是一个回路或是边不相接回路的并。因为每个网孔只包围一个域，所以用其余网孔的环和运算是不能得到这个网孔的。因此，网孔C_1，C_2，…，C_{r-1}是线性独立的。

假设G的任一回路子空间C包围了有限域f_1，f_2，……f_k，那么可以证明

$$C = C_1 \oplus C_2 \oplus \cdots \oplus C_k$$

例如，在图7.4中，集合$C = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\}$包围了域$f_1$，$f_2$，$f_3$，$f_4$。所以，

$$C = C_1 \oplus C_2 \oplus C_3 \oplus C_4$$

因此，G的回路子空间中的每一个元素都能用G中一些或全部网孔的环和来表示。因

为网孔本身是各自独立的，则得如下定理。

定理 7.3 平面图G的所有网孔构成了G的回路子空间的基底。 □

下面的推论是这个定理的直接结果。

推论 7.3.1（欧拉公式） 如果连通平面图G具有m条边、n个顶点和r个域，那么

$$n-m+r=2$$

证明 根据定理7.3，只要注意到，G的零度μ等于$r-1$，上述推论便可得证。 □

通常，检查一个图是否为平面图是很困难的。目前我们利用欧拉公式可以推导出平面图的一些性质。我们将会看到，这些性质能帮助我们在某些情况下，判别非平面性。

推论 7.3.2 如果一个连通的简单平面图G具有m条边和$n\geqslant 3$个顶点，那么

$$m\leqslant 3n-6$$

证明 设$F=\{f_1, f_2, \cdots, f_r\}$表示G域的集合。令域$f_i$的度$d(f_i)$表示$f_i$外界上边的数目，其中桥+记数二次（例如，在图7.4中，域f_1的度为6）。在注意了一个顶点的度和一个域的度之间的类似性后，由定理1.1可得

$$\sum_{f_i\in F}d(f_i)=2m$$

因为图G既无平行边，也无自环，且$n\geqslant 3$，所以对所有i，有$d(f_i)\geqslant 3$。因此

$$\sum_{f_i\in F}d(f_i)\geqslant 3r$$

从而$2m\geqslant 3r$，即

$$r\leqslant\frac{2}{3}m$$

若在欧拉公式中代入这个不等式，则得

$$n-m+\frac{2}{3}m\geqslant 2$$

或

$$m\leqslant 3n-6$$ □

推论7.3.3 K_5是非平面图。

证明 对于K_5，$n=5$和$m=10$。如果它是平面的，那么是推论7.3.2，得

$$m=10\leqslant 3n-6=9$$

这与推论7.3.2相矛盾。因此，K_5一定是非平面的。 □

推论7.3.4 $K_{3,3}$是非平面的。

证明 对于$K_{3,3}$，$m=9$和$n=6$。如果它是平面的，那么根据欧拉公式，它具有$r=9-6+2=5$个域。

在$K_{3,3}$中没有长度小于4的回路，所以每个域的度至少为4，因此

$$2m=\sum_{i=1}^{r}d(f_i)\geqslant 4r$$

+ 桥的定义见习题1.28。——原文注

或

$$r \leqslant \frac{2m}{4}$$

即 $r \leqslant 4$；这与$K_{3,3}$有5个域相矛盾。所以$K_{3,3}$是非平面的。 □

推论7.3.5 在一个简单平面图G中，至少有一个顶点的度小于或等于5。

证明 设G具有m条边和n个顶点。如果G的每个顶点度都大于5，那么根据定理1.1，

$$2m \geqslant 6n$$

或

$$m \geqslant 3n$$

再根据推论7.3.2，

$$m \leqslant 3n-6$$

这两个不等式互相矛盾，所以推论正确。 □

7.3 Kuratowski定理和平面性的另一些特征

本节将介绍由Kuratowski，Wagner，Harary，Tutte和Maclance给出的平面性的特征。

为了解释Kuralowski定理的特征，我们需要定义图之间同胚的概念。

二条与度数为2的顶点相关联的边称为串联边。

设$e_1=(u, v)$和$e_2=(v, w)$是并联在顶点v上的串联边。移去顶点v，并用一条简单边(v, w)代替e_1和e_2，则称为串联合并，见图7.5(a)。



图 7.5 (a)串联合并；(b)串联插入。

在边(u, w)中间插入一个顶点v，而得到边(u, v)和(v, w)，则称为串联插入，见图7.5(b)。

如果两个图是同构的，或者重复使用串联插入和（或）串联合并能使之同构，那么这两个图就称为同胚。

显而易见，如果一个图G是平面的，那么任意与G同胚的图也是平面的，即串联插入或串联合并不影响图的平面性。

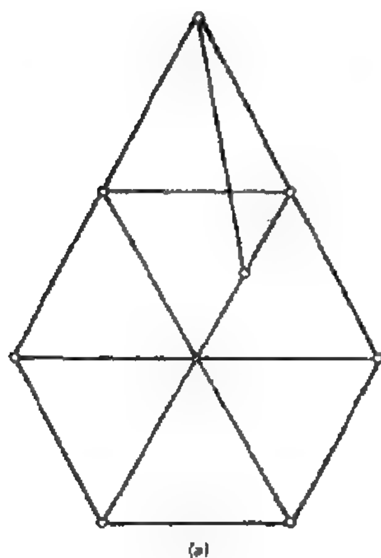
在前节中，我们已经证明了K_5，$K_{3,3}$是非平面图，所以平面图就不包含K_5是$K_{3,3}$同胚子图。值得注意的是，文献[7.5]证明了这个结论的逆也是正确的。在下面的定理中，我们将表述平面性的这个著名特征。它的证明可参见文献[7.6]。

定理7.4（Kuratowski定理） 一个图是平面的，当且仅当它不含有K_5，$K_{3,3}$的同胚子图。 □

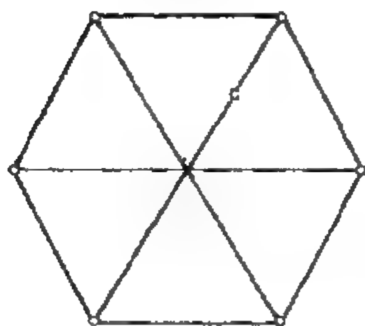
这个定理的证明也可见文献[7.7]。

为了说明Kuratowski定理，考察图7.6所示图G。图G包含了$K_{3,3}$的同胚子图（图7.6(*b*)），所以图G是非平面图。

下面介绍由文献[7.8]和[7.9]单独证明过的平面性的其它特征。



(*a*)



(*b*)

图 7.6 (*a*)图*G*；(*b*)*G*的$K_{3,3}$同胚子图。

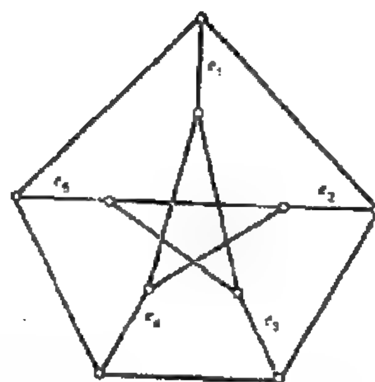


图 7.7 Petersen图

定理7.5 一个图是平面的，当且仅当它不含有可收缩为K_5或$K_{3,3}$的子图。

现在考虑图7.7所示的图（称为Petersen图）。这个图不包含任何同构于K_5或$K_{3,3}$的子图，但已知它是非平面图。所以，如果我们希望用Kuratowski定理来判别Petersen图的非平面性，那么就需要找到K_5或$K_{3,3}$同胚的子图。然而，在压缩掉边e_1，e_2，e_3，e_4和e_5后，这个图就收缩为K_5了，所以这个图的非平面性就很容易从上述定理得出。

下面是Maclane对平面图特性的表述。

定理7.6 一个图是平面的，当且仅当在图G中存在一个基本的回路集合，使没有一个边出现在多于两个回路中。 □

我们知道一个平面图的所有网孔构成了图的回路子空间的一个基，而且图中没有一个边出现在多于两个网孔中。这就证明了定理7.6的必要性。充分性的证明可参见文献[7.10]。

平面图的另外一个重要特征就是存在对偶图。这将在7.5节中讨论。

7.4 对偶图

如果G_2与G_1之间的边存在一一对应关系，并使得G_2中的一个边集是G_2的一个回路矢量，当且仅当G_1中的对应边集是G_1的割集矢量，那么图G_2就是图G_1的对偶。对偶是由文献[7.11]首先定义的，但是他的原始定义是用一种不同形式给出的。

很显然，要证明G_2是G_1的对偶，只需要证明，组成G_2回路子空间的基矢量对应于组成G_1割集子空间的基矢量。

例如，考察图7.8所示图G_1和G_2。G_1的边e_i对应于G_2的边e_i^*。可以证实，回路$\{e_1, e_2, e_3\}$、$\{e_3, e_4, e_5, e_6\}$和$\{e_6, e_7, e_8\}$构成G_1回路子空间的一个基，而且相应的集合$\{e_1^*, e_2^*, e_3^*\}$、$\{e_3^*, e_4^*, e_5^*, e_6^*\}$和$\{e_6^*, e_7^*, e_8^*\}$构成了$G_2$割集子空间的一个基。因此，$G_2$是$G_1$的对偶。

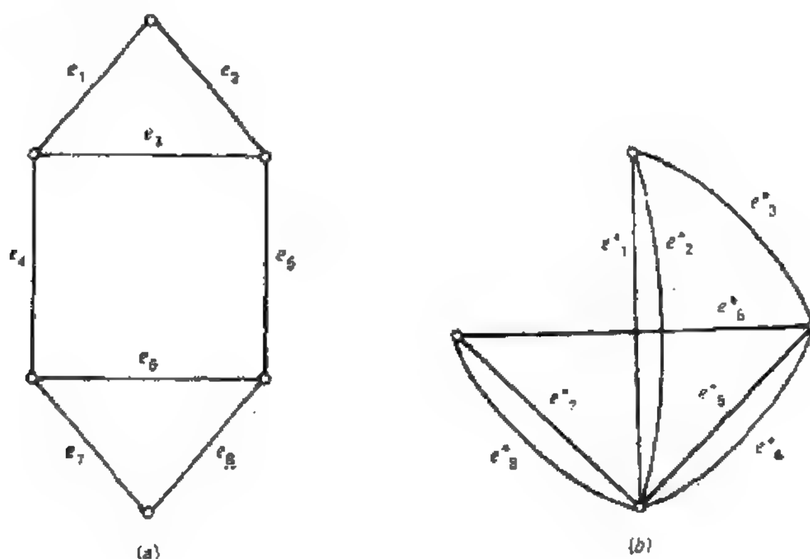


图 7.8 对偶图

(a)图G_1；(b)图G_2。

现在我们研究一个图以及对偶图的一些性质。

定理7.7 设G_2是G_1的对偶，那么G_2中的回路对应于G_1中的割集，反之亦然。

证明 设C^*是G_2的回路，C是在G_1中相应的边集。

假设C不是割集，那么由对偶的定义可得，C必须为不相交的割集C_1，C_2，……C_k的并（$k\geqslant 2$）。

令C_1^*，C_2^*，……C_k^*是G_2中的边集，它对应于割集C_1，C_2，……C_k。再根据对偶的定义可得，C_1^*，C_2^*，……C_k^*是回路，或者是不相交的回路的并。

因为C^*是C_1^*，C_2^*，……C_k^*的并（$k\geqslant 2$），显然C^*必须包含多于一个的回路。但是，这是不可能的，因为C^*是一个回路，而且没有一个回路的真子集是一个回路。因此，$k=1$，或换言之，C是G_1的一个割集。

用类似的方法，我们能证明G_1中的每一个割集对应于G_2中的一个回路。 □

定理7.8 如果G_2是G_1的对偶，那么G_1也是G_2的对偶。

证明 为了证明这个定理，我们需要说明，G_1中的每个回路矢量对应于G_2中的一个割集矢量，反之亦然。

设C是在G_1中的一个回路矢量，而C^*为相应于在G_2中的边集。

根据定理4.7，C与G_1的每个割集矢量都具有偶数条公共边。因为G_2是G_1的对偶，C^*与G_2的每个回路矢量也都具有偶数条公共边。所以，根据定理4.8，C^*是G_2的一个割集矢量。

用类似的方法，我们能证明G_2的每个割集矢量对应于G_1的一个回路矢量。所以，定理成立。 □

鉴于上述定理，如果两个图G_1和G_2中的一个是另一个的对偶，则称图G_1和G_2为简单对偶图。

下面的结论是定理7.8和对偶定义的结果。

定理7.9 如果G_1和G_2是对偶图，那么其中一个的秩等于另一个的零度，即

$$\rho(G_1)=\mu(G_2)$$

或

$$\rho(G_2)=\mu(G_1)$$ □

假设图G有一个对偶，那么就会提出这样一个问题，是否G的每个子图都有一个对偶。为了回答这个问题，我们需要如下的结论。

定理7.10 考虑两个对偶图G_1和G_2。设$e=(v_1, v_2)$是G_1中的一条边，$e^*=(v_1^*, v_2^*)$是G_2中的对应边。设G_1'是在G_1中移去边e后得到的图；G_2'是在G_2中收缩掉边e^*后得到的图。那么G_1'和G_2'是对偶的，它们的边之间同G_1和G_2的一样是一一对应的。

证明 设C和C^*分别表示在G_1和G_2中相应边的集合。

假设C是G_1'中的一个回路。因为C不包含e，所以C也是G_1中的一个回路。因此C^*是G_2中的一个割集，如$\langle V_1^*, V_2^*\rangle$。又因为$C^*$不包含$e^*$，所以顶点$v_1^*$和$v_2^*$都在$V_1^*$或者都在$V_2^*$里。因此$C^*$也是$G_2'$中的一个割集。这样，在$G_1'$中的每一个回路都对应于$G_2'$中的一个割集。

假设C^*是G_2'中的一个割集，因为C^*不包含e^*，它也是G_2中的一个割集。所以C是在G_1中的一个回路。又因为它不包含e，也是G_1'中的一个回路。因此，在G_2'中的每一个割集都相应于G_1'中的一个回路。 □

鉴于这个定理，我们可以用电网络理论的语言来说，图G中的"开路"边对应于G的对偶图中的"短路"边。

下面给出的定理7.10的推论是很有用的。

推论7.10.1 如果图G具有一个对偶，那么G的每一个边-导出子图也都具有一个对偶。

证明 只要我们注意到，G的每个边-导出子图H可以通过移去在G中而不在H中的边获得，这个推论就可根据定理7.10得证 □

为了说明上述推论，考虑图7.8所示的两个对偶图G_1和G_2。图G_1'是通过移去G_1的边e_1和e_8而得到的，如图7.9(a)所示。图G_2'是通过收缩G_2中的e_1^*和e_8^*而得到的，如图7.9(b)

所示。可以证实，G_1'和G_2'也是对偶的。

根据观察，图G中的串联边对应于其对偶图中的并联边，则可得定理7.10的下述推论。

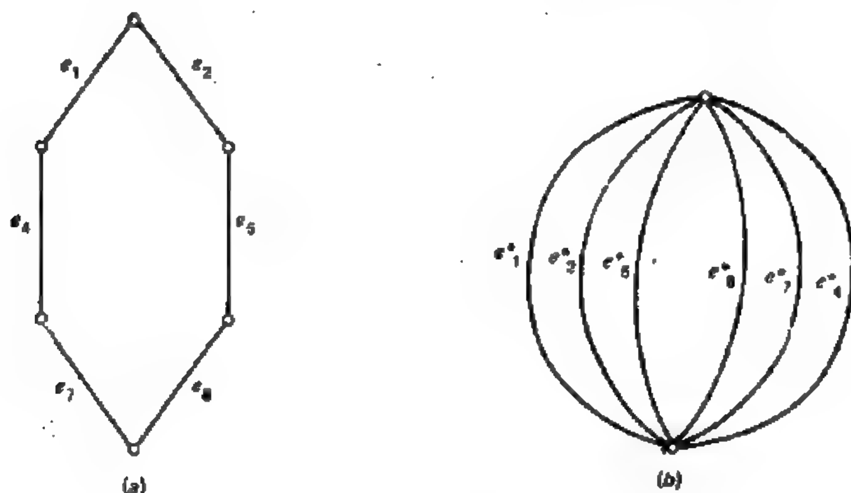


图 7.9 对偶图

(a)图G_1' (b) 图G_2'。

推论7.10.2 如果图G具有对偶，那么每个G的同胚图也具有对偶。 □

我们现在继续叙述对偶的等效特性。

设G为具有n个顶点的图，不失一般性，可令G是连通的。设K^*是G的子图，并设G'是通过收缩K^*的边而得到的图。注意，G'也是连通的。

如果K^*具有n^*个顶点和p个连通片，那么G'就具有$n-(n^*-p)$个顶点。所以，G'的秩为

$$\begin{aligned}\rho(G')&=n-(n^*-p)-1\\&=\rho(G)-\rho(K^*)\end{aligned}\qquad(7.1)$$

定理7.11 设G_1和G_2是两个图，它们之间的边存在一一对应。设H为G_1的任一子图，H^*为G_2中相应的子图。令K^*为G_2中H^*的补图。G_1和G_2是对偶图，当且仅当

$$\mu(H)=\rho(G_2)-\rho(K^*)\qquad(7.2)$$

证明

必要性 设G_1和G_2是对偶图。设G_2'是从G_2中收缩K^*的边后得到的图。那么根据定理7.10，H和G_2'是对偶图，因此，再根据定理7.9，有

$$\mu(H)=\rho(G_2')$$

但根据（7.1）式，有

$$\rho(G_2')=\rho(G_2)-\rho(K^*)$$

因此

$$\mu(H)=\rho(G_2)-\rho(K^*)$$

充分性 假设（7.2）式满足G_1的每个子图H。现在我们可以证明G_1中的每个回路都对应于G_2中的一个割集，反之亦然。

设H为G_1中的回路。因此，$\mu(H)=1$。所以根据（7.2）式，有

$$\rho(K^*)=\rho(G_2)-1$$

因为H是G的零度为1的最小子图，而且K^*是G_2中H^*的补图，所以显而易见，K^*是G_2的秩为$\rho(G_2)-1$的最大子图。根据割集的定义，可证明H^*是G_2中的一个割集。

用类似的方法，我们能证明G_2中的一个割集对应于G_1中的一个回路。

因此，G_1和G_2是对偶图。 □

文献[7.11]关于对偶性的原始定义，与定理7.11的叙述一样。

为了说明这个定义，考虑图7.8所示对偶图G_1和G_2。G_1的子图H和G_2中对应子图的补图K^*，如图7.10所示。现在可以证实

$$\mu(H)=\rho(G_2)-\rho(K^*)$$

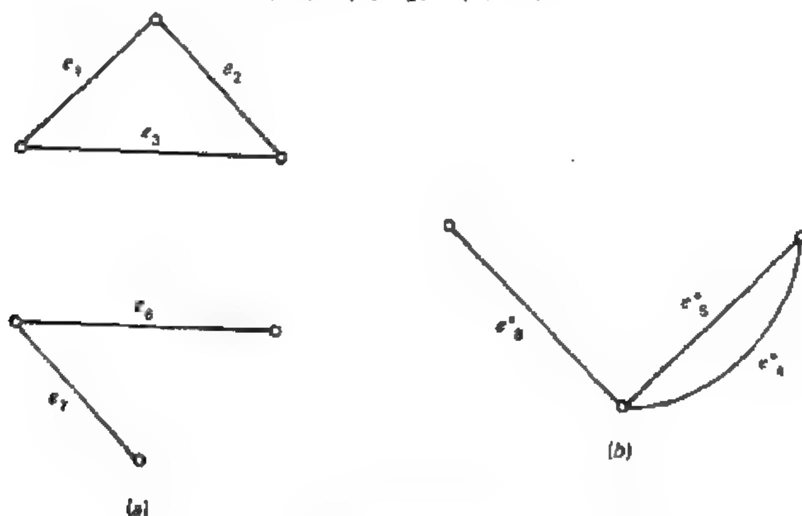


图 7.10 Whitney对偶性定义的说明
(a)图H，$\mu(H)=1$；(b)图K^*，$\rho(K^*)=2$。

7.5 平面性和对偶性

本节将给出这样一类图的特性，它们都具有对偶图。这样做是为了把二个似乎没有联系的概念（平面性和对偶性）联系起来。

首先，我们要证明每个平面图都具有对偶图。这个证明是基于构造已知平面图的对偶过程。

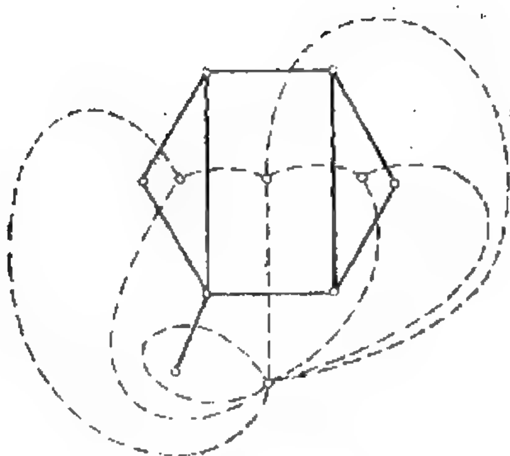
考虑一个平面图，并设G为这个平面图的可平面嵌入。设f_1，f_2，…，f_r为G的域。按下述定义来构造一个图G^*：

1.G^*有r个顶点v_1，v_2，…，v_r，顶点v_i对应于域f_i，$1\leqslant i\leqslant r$。

2.G^*和G有相同的边数。

3.如果G的一条边e是域f_i和f_j的公共边（不必一定是相异的），那么G^*中相应的边e^*连接顶点v_i^*和v_j^*（注意，G中的每个边e最多只是两个域的公共边，并且一条边仅仅在一个域内也是可能的）。

构造G^*的另一个简单方法是：首先将顶点v_1^*，v_2^*，……，v_r^*放在图G的每个域内。然后，对于域f_i和f_j的每个公共边e，画一条连接v_i^*和v_j^*的线，并通过边e。这个线就代表


图 7.11 对偶图的构造

边e^*。

构造G^*的过程用图 7.11 来说明。实线代表已知平面图G的边，虚线表示G^*的边。

我们现在来证明G^*是G的对偶。

设C_1，C_2，……C_{r-1}代表G的网孔，C_1^*，C_2^*，………C_{r-1}^*代表G^*中对应的边集。显且易见，根据构造G^*的过程，C_i^*中的边都与顶点v_i^*相关联，并形成了一个切割，若移去这些边，将使v_i^*从G^*的剩余顶点中分离出来。

根据定理7.3，C_1，C_2，…，C_{r-1}形成了G的回路子空间的一个基，并且还知道，关联矢量C_1^*，C_2^*，…，C_{r-1}^*形成了G^*的割集子空间的一个基。因为在C_i和C_i^*的矢量之间存在一一对应，所以G和G^*是对偶图。因此，我们有下述定理。

定理7.12 每一个平面图都有一个对偶。 □

紧接着会提出这样一个问题，非平面图是否也有对偶。回答是“没有”，这个回答是基于下面的两个引理。

引理7.1 $K_{3,3}$无对偶。

证明 首先观察到：

1.$K_{3,3}$没有两条边的割集。

2.$K_{3,3}$仅有长度为 4 或 6 的回路。

3.$K_{3,3}$有 9 条边。

假设$K_{3,3}$有对偶G，那么这些观察将分别意味着G为

1.G没有二条边的回路，即G没有平行边。

2.G没有小于 4 条边的割集。因此，G中的每个顶点的度至少为 4 。

3.G有 9 条边。

上面头二条意味着G至少有 5 个顶点，每个顶点的度至少为 4 。因此，G必须至少有$\frac{1}{2}\times 5\times 4=10$条边。然而，这于第三条矛盾。所以，$K_{3,3}$没有对偶。 □

引理7.2 K_5无对偶。

证明 首先观察到：

1.K_5没有长度为 1 或 2 的回路。

2.K_5仅有 4 条边或 6 条边的割集。

3.K_5有10条边。

假设K_5有对偶G，那么用上面观察到的第二条，G仅有长度为 4 或 6 的回路。换言之，G的所有回路长度都为偶数。所以G是二分的。

因为一个具有 6 个或更少顶点的二分图，不可能有 9 条以上的边，否则G必须要有 7

个顶点。但是根据观察的第一条，G的每个顶点的度至少为 3，所以 G 必须至少具有 $\frac{1}{2}\times7\times3>10$条边。然而，这和观察到的第三条相矛盾，所以$K_5$没有对偶。 □

本节主要结论如下。

定理7.13 一个图有对偶，当且仅当它是平面的。

证明 这个定理的充分性和定理7.12一样。

我们通过说明一个非平面图G没有对偶的方法来证明必要性。根据Kuratowski定理，G有一个和$K_{3,3}$或K_5同胚的子图H。如果G有对偶，那么根据推论7.10.1，H具有对偶。但是，又根据推论7.10.2，$K_{3,3}$或者K_5应该有对偶。然而，这将与$K_{3,3}$和K_5没有对偶相矛盾。所以G没有对偶。 □

以上定理给出了平面图的特性，即平面图具有对偶图，这最早是由whitney证明的。而这里的证明是由文献[7.12]给出的。whitney的原始证明没有使用Kuratowski定理，可参见文献[7.13]。

根据本节中早已给出的证明过程，很显然，一个平面图的不同的（虽然同构）平面嵌入可以导致非同构的对偶（习题7.6）。下面的定理介绍了一个图的对偶性质。

定理7.14 一个图G的所有对偶都是2-同构的；每个与G的对偶2-同构的图也是G的对偶。 □

根据对偶定义和定理1.7，上述定理可用直接的方法证明。

7.6 进一步阅读

文献[7.4]、[7.11]、[7.14]，以及由文献[7.13]、[7.15]、[7.6]、[7.16]写成的书，都特别值得推荐的进一步阅读资料。至于判定图的平面性算法，可见文献[7.17]。

非平面图G的两个感兴趣的性质是：

1.平面子图的最小数目称为G的厚度，而由这些平面子图组成的并为G。

2.为了在平面上画出一个图因而出现交叉（或相交）的最小数目，称为G的交叉数。

关于非平面图的厚度和交叉数的一些结论，可参见文献[7.6]，也可参看文献[7.18]。

7.7 习　题

7.1 试证：如果G是具有m条边、n个顶点和围长$k\geqslant3$（围长定义见习题1.18）的连通平面图，那么$m\leqslant k(n-2)/(k-2)$，并用这个结论推出Petersen图是非平面的。

7.2 证明：一个具有$n\geqslant4$个顶点的简单图至少有4个顶点，每个顶点的度数等于或小于5。

7.3 证明或否定：每个连通的简单非平面图都可收缩为K_5或$K_{3,3}$。

7.4 设G是具有至少十一个顶点的简单图。试证G和它的补图$\overline{G}$都不是平面图。（事实上，用九个顶点代替十一个顶点，就可证明同样的结论。见文献[7.19]）试给出一个八个顶点图G的例子，而G具有和$\overline{G}$都是平面图的性质。

7.5 利用Kuratowski定理，证明Petersen图是非平面图。

7.6 找出图7.12所示图的二个非同构的对偶图。

7.7 证明：一个无自环的平面图是不可分的，当且仅当他的对偶是不可分的。

7.8 证明：不可分的平面图的对偶是欧拉图，当且仅当这个图是二分图。

7.9 一个平面图和它的对偶同构时，则称它为自对偶。证明：如果图G是具有n个顶点，m条边的自对偶，那么$m=2n-2$。

7.10 有两个顶点指定为端点的图，称为二端对图。若二端对图是平面的，而且把一条连接两个指定端点的边加上后所得到的图仍是平面的，这样的图称为平面二端对图。

串并图是下面递归定义的二端对图：

（a）单独的边是一个串并图，如果G'和G''是串并的，那么：

（b）G'和G''的串联组合是一个串并图。G'和G''的串联组合就是把G'的一个端点和G''的一个端点连接起来（图7.13(a)）。

（c）G'和G''的并联组合是一个串并图。G'和G''的并联组合就是把G'的两个端点和G''的两个端点分别连接起来（图7.13(b)）。

证明：图G的对偶图是一个串并图，当且仅当G是串并的。

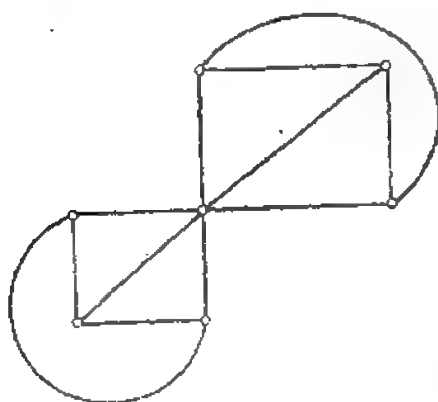


图 7.12

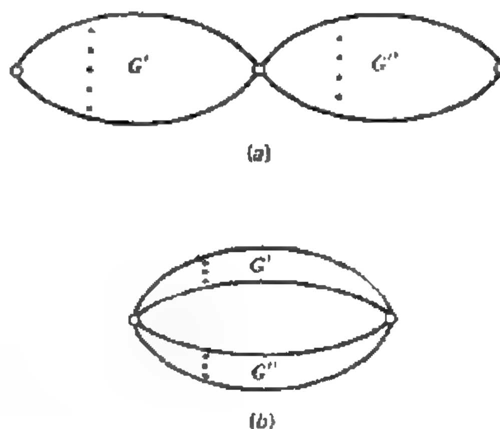


图 7.13 (a)G'和G''的串联组合；(b)G'和G''的并联组合。

7.8 参考文献

7.1 K. Wagner, "Bemerkungen zum Vierfarbenproblem," *Über. Deutsch Math. Verein*, Vol. 46, 26—32(1936).

7.2 I. Fary, "On Straight Line Representation of Planar Graphs," *Acta Sci. Math. Szeged*, Vol. 11, 229—233(1948).

7.3 S. K. Stein, "Convex Maps," *Proc. Am. Math. Soc.*, Vol. 2, 464—466(1951).

7.4 H. Whitney, "Non-Separable and Planar Graphs," *Trans. Am. Math. Soc.*, Vol. 34, 339—362(1932).

7.5 C. Kuratowski, "Sur le probleme des courbes gauches en topologie," *Fund. Math.*, Vol. 15, 271—283(1930).

7.6 F. Harary, *Graph Theory*, Addison-Wesley, Reading, Mass., 1969.

7.7 W. T. Tutte, "How to Draw a Graph," *Proc. London Math. Soc.*, Vol. 13, 743—767(1963).

7.8 K. Wagner, "Über eine Eigenschaft der ebenen Komplexe," *Math Ann.*, Vol. 114, 570—590(1937)

7.9 F. Harary and W. T. Tutte "A Dual Form of Kuratowski's Theorem," *Canad. Math. Bull.*, Vol. 8, 17—20(1965)

7.10 S. MacLane, "A Structural Characterization of Planar Combinatorial Graphs," *Duke Math. J.*, Vol 3, 340—372(1937).

7.11 H. Whitney, "Planar Graphs," *Fund. Math.*, Vol 21, 73—84(1933).

7.12 T. D. Parsons, "On Planar Graphs," *Am. Math. Monthly*, Vol. 78, 176—178(1971).

7.13 S. Seshu and M. B. Reed, *Linear Graphs and Electrical Networks*, Addison-Wesley, Reading, Mass., 1961.

7.14 H. Whitney, "A Set of Topological Invariants for Graphs," *Am. J. Math.*, Vol. 55, 231—235(1933).

7.15 O. Ore, *The Four Colour Problem*, Academic Press, New York, 1967.

7.16 J. A. Bondy and U. S. R. Murty, *Grcph Theory with Applications*, Macmillan, London, 1976.

7.17 J. E. Hopcroft and R. E. Tarjan, "Efficient Planarity Testing," *J. ACM*, Vol. 21, 549—568(1974).

7.18 N. K. Bose and K. A. Prabhu, "Thickness of Graphs with Degree Constrained Vertices," *IEEE Trans. Circuits and Sys.*, Vol. CAS-24, 184—190(1975).

7.19 W. T. Tutte, "On the Nonbiplanar Character of the Complete 9-graph," *Canad. Math. Bull.*, Vol. 6, 319—330(1963).

第八章　连通度和匹配

在第一章中我们定义了一个图是连通的，如果图中任意两顶点之间都存在着一条路径。假定一个图G是连通的，那么我们所感兴趣的是发现G的"连通程度如何"，换言之，我们想要知道移去那些使图变为不连通的顶点或边的最小数目。这样，就引出了一个图的顶点和边的连通度概念。在本章中，我们首先导出几个与图的顶点和边的连通度有关的结论，并讨论图论中一个经典的结论，即Menger定理，它将使顶点-不相接和边-不相接的路径与连通度联系起来。

一个图中的匹配是一个边的集合，其中任意两条边没有公共顶点。在本章的后半部分，我们从Hall结婚定理开始推导出匹配的理论。

连通度和匹配是图论中两个极为有研究价值的课题。图论中许多较深的结论都是属于这两个方面的。

8.1　连通度或顶点连通度

一个图G的连通度$\kappa(G)$是使图G变为不连通图或平凡图[1]而移去的最小顶点数目，$\kappa(G)$也称为顶点连通度，以区别于边连通度。在下一节中将介绍边连通度。

例如，图8.1所示的图的连通度是2，因为从图中移去顶点v_1和v_2就产生了一个不连通图，而不论移去任一单独顶点都不能得到一个不连通图。

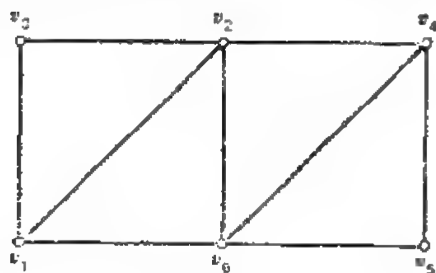


图 8.1

很显然，一个不连通图的连通度为零。

考虑一个具有n个顶点的图。如果G是完全图，很显然$\kappa(G)=n-1$。如果G是非完全图，则它至少有两个不邻接的顶点v_1和v_2。若从G中移去其余$n-2$个节点，则留下了一个仅含有v_1和v_2的不连通图。这样，如果图G是非完全图，所以有

$$\kappa(G)\leq n-2$$

图G的一个不连通集是一个顶点集合，若从G中移去这个集合将产生一个不连通图或平凡图。

一个图G被称为是k-连通的，如果$\kappa(G)\geq k$。所以，一个k-连通图不存在不连通集S，其中$|S|\leq k-1$。

如果一个图是连通的，则它的连通度大于或等于1。因此，连通图是1-连通的。

[1] 1.7节介绍过仅含单一顶点的图称为平凡图。——原文注

如果一个连通图没有切割-顶点，那么它的连通度大于1。因此，没有切割-顶点的连通图是2-连通的。

在下面的定理中，我们将介绍一个图的连通度的直观上界。

定理8.1 对于一个简单的连通图G，$\kappa(G)\leqslant\delta(G)$，这里$\delta(G)$是$G$中的最小度。

证明

考虑一个简单连通图$G=(V, E)$中的任意顶点v。令$\Gamma(v)$表示与v相邻接的顶点集。显然，$\Gamma(v)$是一个连通集，这是因为移去$\Gamma(v)$中的顶点将产生一个平凡图或不连通图，而且图中v不跟任意一个剩余顶点相连接。这样

$$\kappa(G)\leqslant|\Gamma(v)|\text{，对于所有}v\in V$$

因为G是简单的，则有

$$|\Gamma(v)|=d(v)$$

从而

$$\kappa(G)\leqslant\min_{v\in V}\{d(v)\}=\delta(G)$$

□

如果一个图G具有m条边和n个顶点v_1，v_2，…，v_n，那么由定理1.1可知，

$$d(v_1)+d(v_2)+\cdots+d(v_n)=2m$$

因此

$$n\cdot\delta(G)\leqslant 2m$$

$$\delta(G)\leqslant\left\lfloor\frac{2m}{n}\right\rfloor \tag{8.1}$$

这里$\lfloor x\rfloor$表示小于或等于x的最大整数。

若将（8.1）式与定理8.1的结论联立起来，则得如下定理。

定理8.2 对于一个具有m条边n个顶点的简单连通图G，

$$\kappa(G)\leqslant\left\lfloor\frac{2m}{n}\right\rfloor$$

□

令$f(k, n)$表示n个顶点的k-连通图所必须具有的最少边数。当然，我们假定$k<n$。由定理8.2，显然有

$$f(k, n)\geqslant\left\lceil\frac{kn}{2}\right\rceil \tag{8.2}$$

这里$\lceil x\rceil$表示大于或等于x的最小整数。

文献[8.1]已经证明并给出了构造一个恰好具有$\lceil kn/2\rceil$条边的k-连通图$H_{k,n}$的步骤，从而使（8.2）式等号成立。这个步骤如下：

情形1 k是偶数。

令$k=2r$，则$H_{2r,n}$具有顶点v_0，v_1，v_2，…，v_{n-1}。如果$i-r\leqslant j\leqslant i+r$，则两个顶点$v_i$和$v_j$是邻接的，这里加法是模$n$运算。

$H_{6,8}$如图8.2(a)所示。

情形2 k是奇数，n是偶数。

令$k=2r+1$。$H_{2r+1,n}$是这样构造的，首先构造出$H_{2r,n}$，然后对于$i=1$，2，…，$n/2$，

再从顶点v_i到v_j添加一条边，并且$j=i+n/2$（模n运算）。

$H_{5,8}$如图8.2(b)所示。

情形3　k是奇数，n是奇数。

令$k=2r+1$。$H_{2r+1,n}$是这样构造的，首先构造出$H_{2r,n}$，然后对于$i=0$，1，2，…，$(n-1)/2$，再从v_i到v_j添加一条边，并且$j=i+(n+1)/2$（模n运算）。

$H_{7,9}$如图8.2(c)所示。

很容易验证上面构造的图$H_{k,n}$恰好具有$\lceil kn/2\rceil$条边。下面我们来证明$H_{k,n}$是k-连通的。

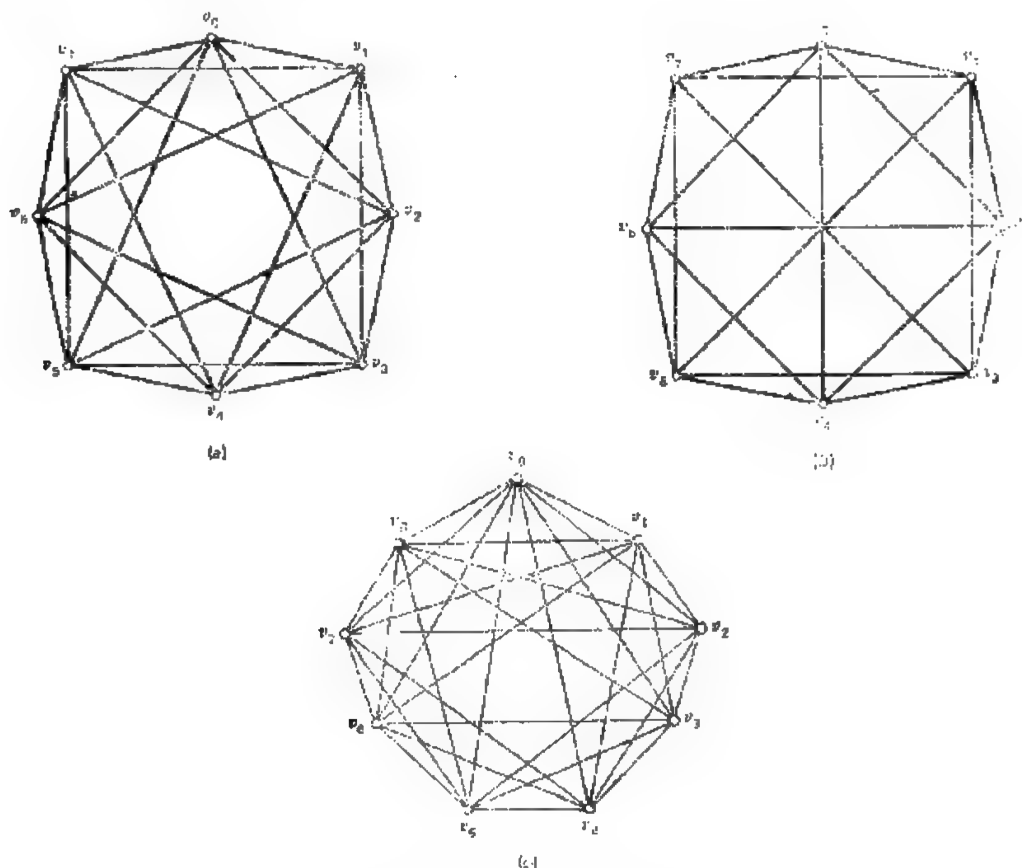


图8.2　(a)$H_{6,8}$　(b)$H_{5,8}$　(c)$H_{7,9}$

定理8.3　图$H_{k,n}$是k-连通的。

证明

情形1　$k=2r$。

由于$H_{2r,n}$的对称性，这就足以证明：通过移去少于$2r$个顶点就不可能使顶点v_0和v_α不连通，其中$\alpha=1$，2，…，$n-1$。假定移去$2r-1$个顶点v_{i_1}，v_{i_2}，…，$v_{i_{2r-1}}$，就可使v_0和v_α不连通，两个间隔$[0,\ \alpha]$、$[\alpha,\ n]$之一至多含有$r-1$个这些下标。假定这些下标是在间隔$[0,\ \alpha]$中，那么从序列v_0，v_1，v_2，…，v_α中移去v_{i_1}，v_{i_2}，…，$v_{i_{2r-1}}$所得到的两个连续顶点是由一条边连通的（因为它们下标之差$\leqslant r$）。因此，从v_0到v_α一定存在一条路径，这与前面相矛盾。

情形 2　$k=2r+1$，n为偶数。

假定移去$2r$个顶点v_{i_1}，v_{i_2}，…，$v_{i_{2r}}$后，v_0和v_α是不连通的。如果两个间隔$[0,\alpha]$、$[\alpha, n]$之一不含有这些下标中的连续r个数，那么可构造一条从v_0到v_α的路径，如情形1所示。因此，假定移去v_i，v_{i+1}，…，v_{i+r-1}后，v_0和v_α是不连通的，这里$1\leqslant i\leqslant \alpha-r$，以及$v_{\alpha+j}$，$v_{\alpha+j+1}$，…，$v_{\alpha+j+r-1}$，这里$1\leqslant j\leqslant n-r-\alpha$。

令　$\beta=\left[\dfrac{\alpha+i+j+r+n-1}{2}\right]$，$\beta'=\beta+\dfrac{n}{2}$（模$n$）

那么

$$\beta\in[\alpha+j+r,\ n+i-1],\ \beta'\in[i+r,\ \alpha+j-1]。$$

从顶点v_0到v_β和从$v_{\beta'}$到v_α都存在一条路径。因为在$H_{2r+1,n}$中存在一条从v_β到$v_{\beta'}$的边，因此，从顶点v_0到v_α存在一条路径。

情形 3　$k=2r+1$，n为奇数。

证明同情形 2。 □

在下面的定理中，我们介绍一个图是k-连通的充分条件。这个结论见文献[8.2]。

定理8.4　设G是一个n阶简单图。假如我们指定G的顶点为：

$$d(v_1)\leqslant d(v_2)\leqslant\cdots\leqslant d(v_n)$$

如果

$$d(v_r)\geqslant r+k-1,\qquad 对于1\leqslant r\leqslant n-1-d(v_{n-k+1})$$

那么G就是k-连通的。

证明

假定简单图G满足定理的条件。如果简单图G不是k-连通的话，那么就存在一个不连通集S，使得$|S|=s<k$。

现在考虑不连通图$G-S$。令H是具有最小阶h的$G-S$的一小片。那么很显然，H中每个顶点度至多为$h-1$。因此，在G中，H的每个顶点的度至多为$h+s-1$。这样

$$d(v_h)\leqslant h+s-1<h+k-1 \tag{8.3}$$

因此，由定理的条件，

$$h>n-1-d(v_{n-k+1}) \tag{8.4}$$

因为$G-S$具有$n-s$个顶点，而且H是具有最小阶的$G-S$的片，所以有

$$h\leqslant n-s-h$$

或

$$h+s\leqslant n-h$$

因此，

$$d(v)\leqslant h+s-1\leqslant n-h-1,\qquad v\in V(H) \tag{8.5}$$

这里$V(H)$是H的顶点集。因为每个顶点$u\in V(G)-V(H)-S$至多与$n-h-1$个顶点是相关联的，所以有

$$d(u)\leqslant n-h-1,u\in V(G)-V(H)-S \tag{8.6}$$

由（8.5）和（8.6）式可以得出：度数超过$n-h-1$的所有顶点一定在S中。这样至多有S个度数超过$n-h-1$的顶点。因此

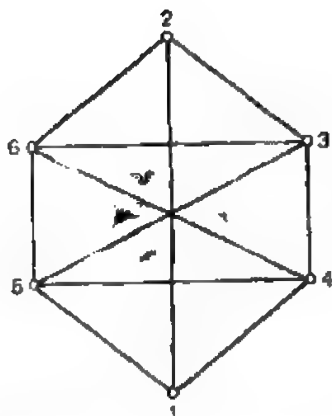


图 8.3 满足定理8.4、$k=3$ 的条件的 3-连通图

$$d(v_{n-s})\leqslant n-k-1 \tag{8.7}$$

应用（8.4）式和（8.7）式，我们得到

$$d(v_{n-s})<d(v_{n-k+1})$$

因此

$$n-s<n-k+1$$

或

$$s\geqslant k$$

这与$s<k$出现矛盾。 □

例如，图 8.3所示图的度数满足定理 8.4$k=3$的条件，所以它是3-连通的。

8.2 边连通度

一个图G的边连通度$\kappa'(G)$是一个使图变为不连通图或平凡图，并在G中需移去的最小边数。换言之，$\kappa'(G)$是一个具有最小边数切割中的边数。例如，对于图 8.4所示的图，边连通度等于 2，这是因为移去两条边e_1和e_2就使图不连通，而移去任何单个边都不可能得到不连通图。

如果$\kappa'(G)\geqslant k$，图G称作k-边-连通的。这样，至少要移去k条边才可使一个k-边-连通图成为不连通的。

因为关联在G中任何一个顶点上的边形成了G的一个切割，则得

$$\kappa'(G)\leqslant\delta(G)$$

在下面的定理中，我们给出$\kappa(G)$、$\kappa'(G)$和$\delta(G)$之间的关系。

定理8.5 对于一个简单图G，

$$\kappa(G)\leqslant\kappa'(G)\leqslant\delta(G)$$

证明

我们已经证明过了第二个不等式。第一个不等式证明如下。

如果G是不连通的，那么$\kappa(G)=\kappa'(G)=0$。在这种情况下，条件$\kappa(G)\leqslant\kappa'(G)$成立。

如果G是连通的，并且$\kappa'(G)=1$，那么G具有一个桥e。在这种情况下，如果我们移去与e关联的一个顶点，那么则产生一个不连通图或平凡图。这样，$\kappa(G)\leqslant\kappa'(G)$在这种情况下也成立。

假定$\kappa'(G)\geqslant 2$，那么G有$\kappa'(G)$条边，若移去这些边，使得它成为不连通图。若移去这些边中的任意$\kappa'(G)-1$条边，都会产生一个具有桥$e=(v_1, v_2)$的图。这$\kappa'(G)-1$条边中的每条边都选择一个不同于v_1和v_2的端顶点，若移去这些顶点，将要从G中移去$\kappa'(G)-1$条边，甚至可能移去更多的边。假定所得到的图是不连通的，那么$\kappa(G)\leqslant\kappa'(G)-1$。否则这个图将有一条边$e$作为桥，因此移去$v_1$或$v_2$将产生一个不连通图或平凡图。在这种情况下，$\kappa(G)\leqslant\kappa'(G)$。

这样在所有的情况下，

$$\varkappa(G) \leqslant \varkappa'(G)$$ □

下面我们介绍$\varkappa'(G)$等于$\delta(G)$的充分条件，这个结论见文献[8.3]。

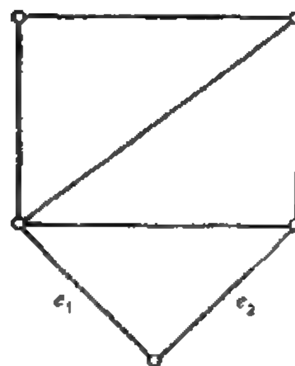


图 8.4　2-边-连通图

定理8.6　设G是n-顶点简单图。如果$\delta(G) \geqslant \lfloor n/2 \rfloor$，那么$\varkappa'(G)=\delta(G)$。

证明

人们可以证明G是连通的（见习题1.14），因此$\varkappa'(G)>0$。

由于$\varkappa'(G) \leqslant \delta(G)$，如果我们证明了$\varkappa'(G) \geqslant \delta(G)$，那么就可证得定理。

假定$\varkappa'(G)<\delta(G)$，那么就存在着一个切割$S=\langle V_1, \overline{V}\rangle$，使得$\varkappa'(G)=|S|<\delta(G)$。令$S$的边与$V_1$中的$q$个顶点和$\overline{V}_1$中的$p$个顶点相关联。

假定$|V_1|=q$，那么V_1的每个顶点至少是S中一条边的端点。如果我们用G_1表示由顶点集V_1诱导出的G的子图，那么G_1至少有

$$m_1=\frac{1}{2}(q\delta(G)-\varkappa'(G))$$

条边。因为$\varkappa'(G)<\delta(G)$，所以有

$$\begin{aligned} m_1 &> \frac{1}{2}(q\delta(G)-\delta(G)) \\ &= \frac{1}{2}\delta(G)(q-1) \\ &> \frac{1}{2}q(q-1) \end{aligned}$$

因为$\delta(G)>\varkappa'(G) \geqslant q$。这与假定相矛盾，因为在一个简单图中，不可能有多余$q(q-)/2$条边联接q个顶点。因此，$|V_1|>q$。以同样的方法，我们可以证明$|\overline{V}_1|>p$。

如果$|V_1|>q$且$|\overline{V}_1|>p$，那么在V_1和$\overline{V}$中仅存在着分别与V_1和$\overline{V}_1$中的顶点相邻接的顶点。这样，V_1和$\overline{V}_1$都至少含有$\delta(G)+1$个顶点。因此G至少具有$2\delta(G)+2$个顶点。但是

$$\begin{aligned} 2\delta(G)+2 &\geqslant 2\cdot\left\lfloor\frac{n}{2}\right\rfloor+2 \\ &> n \end{aligned}$$

这就导致了矛盾。因此不存在$|S|<\delta(G)$的切割S。 □

8.3　规定度的图

曾记得，如果存在着一个n-顶点图，其顶点$v_1, v_2, \cdots, v_n$使得v_i具有度d_i，那么一个非负整数序列（$d_1, d_2, \cdots, d_n$）是可以用图来表示的。

本节中，我们首先描述一个构造简单图的算法，如果这个图存在的话，那么这个图就具有一个规定的度序列。然后我们利用这个算法来建立基于具有规定度序列的k-连通图

存在的Edmond定理。

考虑一个图序列（d_1, d_2, …, d_n），其中 $d_1 \geqslant d_2 \geqslant \cdots \geqslant d_n$。令 d_i 是顶点 v_i 的度。"划出 d_k" 意谓着把相应的顶点 v_k 连接到顶点

$$v_1,\ v_2,\ \cdots,\ v_{d_k}, \quad 如果 d_k < k$$

或者连接到顶点

$$v_1,\ v_2,\ \cdots,\ v_{k-1},\ v_{k+1},\ \cdots,\ v_{d_k+1}, \quad 如果 \geqslant k$$

序列

$$(d_1-1,\ \cdots,\ d_{d_k}-1,\ d_{d_k+1},\ \cdots,\ d_{k-1},\ 0,\ d_{k+1}\cdots,\ d_n),\quad 如果 d_k<k$$

或

$$(d_1-1,\ \cdots,\ d_{k-1}-1,\ 0,\ d_{k+1}-1,\ \cdots,\ d_{d_k+1}-1,\ d_{d_k+2},\ \cdots,\ d_n),\quad 如果 d_k \geqslant k$$

称为划出 d_k 后的剩余序列，或简称剩余序列。

文献[8.4]和[8.5]已经给出了构造一个简单图的算法，如果简单图存在的话，其条件是具有一个规定度序列。这种算法是建立在下面定理的一个特例($k=1$)基础上的，下面定理是在文献[8.6]中提出的。

定理8.7 如果一个序列（d_1, d_2, …, d_n）且 $d_1 \geqslant d_2 \geqslant \cdots \geqslant d_n$ 是一个简单图的度序列，那么也就是划出 d_k 后的剩余序列。

证明

为了证明定理，我们必须证明一个以（d_1, d_2, …, d_n）作为其序列的图存在，并使得顶点 v_k 与除它自己以外最先的 d_k 个顶点相邻接。若不然，我们从那些具有度序列（d_1, d_2, …, d_n）的图中选出一个简单图 G，其中 v_k（除自身之外）与最先的 d_k 个顶点中的最大数的顶点相邻接。设 v_m 是一个不与 G 中 v_k 相邻接的顶点，以致

$$m \leqslant d_k, \quad 如果 d_k < k$$

或者

$$m \leqslant d_{k+1}, \quad 如果 k \leqslant d_k$$

换言之，v_m 是在最先的 d_k 个顶点中，但不是在 v_k 中。所以 v_k 在 G 中与某一个不在最先的 d_k 个顶点中的顶点 v_q 相邻接。其次 $d_m > d_q$（如果相等，则 q 和 m 的次序可以相互交换）。因而 v_m 与某个顶点 v_t（$t \neq q$, $t \neq m$）相邻接，以使得 v_t 和 v_q 在 G 中不邻接。如果我们移去边（v_m, v_t）和（v_k, v_q），并用（v_m, v_k）和（v_t, v_q）来替换之，我们就可得到一个图 G'，G' 中有多于一个顶点（除自身以外）与最先的 d_k 个顶点中的 v_k 相邻接，G' 违背了 G 的定义。 □

从上面的定理我们得到了下面的算法，它是由一个简单图来实现一个序列 $D=(d_1, d_2, \cdots, d_n)$，且 $d_1 \geqslant d_2 \cdots d_n$ 的Hakimi算法的推广。

选择任何 $d_k \neq 0$，将 v_k（除自身之外）与最先的 d_k 个顶点相连接，这样就"划出" d_k。为了计算剩余度序列，重新排列顶点，使得所产生的序列中的剩余度以非增顺序排列。重复这一过程，直到下面的情况之一发生：

1. 所有余度为0，在这种情况下，产生的图以 D 作为它的度序列。
2. 余度中有一个是负的。这就意谓着序列 D 不能用图表示。

为了说明上面的算法，考虑序列

$$\begin{matrix} & v_1 & v_2 & v_3 & v_4 & v_5 \\ D=(& 4 & 3 & ③ & 2 & 2) \end{matrix}$$

在找出d_3（加圆圈）之后，我们得到序列

$$\begin{matrix} & v_1 & v_2 & v_3 & v_4 & v_5 \\ D'=(& 3 & 2 & 0 & 1 & 2) \end{matrix}$$

在对余度重新排序之后，上式就变成

$$\begin{matrix} & v_1 & v_2 & v_5 & v_4 & v_3 \\ D_1=(& 3 & 2 & ② & 1 & 0) \end{matrix}$$

我们再找出与v_5相对应的度，并且得到

$$\begin{matrix} & v_1 & v_2 & v_5 & v_4 & v_3 \\ D_1'=(& 2 & 1 & 0 & 1 & 0) \end{matrix}$$

在D_1'中对余度重新排序，得到

$$\begin{matrix} & v_1 & v_2 & v_4 & v_5 & v_3 \\ D_2=(& ② & 1 & 1 & 0 & 0) \end{matrix}$$

现在划出对应于v_1的度，我们得到

$$\begin{matrix} & v_1 & v_2 & v_4 & v_5 & v_3 \\ D_2'=(& 0 & 0 & 0 & 0 & 0) \end{matrix}$$

算法到此结束。因为所有的余度都等于0，所以序列（4，3，3，2，2，）是可用图表示的。所需的图（图8.5）可由下面的步骤获得，它对应于度被划出的次序：

1. 连接v_3到v_1，v_2和v_4。

2. 连接v_5到v_1和v_2。

3. 连接v_1到v_2和v_4。

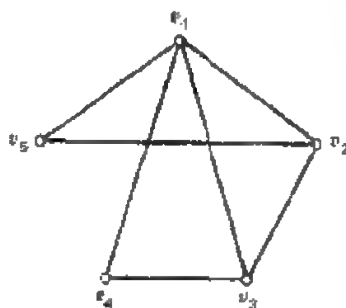


图8.5 具有度序列为（4，3，3，2，2）的图

文献[8.7]给出了一个序列可用图表示的主要条件（不是算法形式），也可看文献[8.8]。

假定在上面的算法中，我们在每一步划中都给出了最小的非零余度，然后通过归纳就可很容易地证明，如果

$$d_i \geq 1, \quad 对于所有 i \tag{8.8}$$

以及

$$\sum_{i=1}^{n} d_i \geq 2(n-1) \tag{8.9}$$

那么产生的图就是连通的。注意（8.8）式和（8.9）式是连通图的必要条件。事实上，在下面的定理中证明了一个更有说服力的结论。这个结论是由文献[8.9]提出的，证明是由文献[8.10]给出的。

定理8.8（EDMONDS） 一个图序列（d_1，d_2，…，d_n）是一个简单k-边-连通图

的度序列（$k \geq 2$）的充要条件是每个度$d_i \geq k$。

证明

必要性是显然的。

下面我们通过证明当给定的图序列中的度大于或等于k时，我们刚才新描述的算法将产生一个k-边-连通图，以此来证明定理的充分性。注意在算法的每一步中，我们都应该划出最小的非零余度。

用归纳法来证明。假定对于每个度$d_i \geq p$且$p \leq k-1$的所有序列，算法是成立的。

为了证明定理，我们必须证明在由算法所构造的图中，每个割集$\langle A, \bar{A} \rangle$至少有$k$条边。如果$|A| = 1$或$|\bar{A}| = 0$，则证明是明显的。所以假定$|A| \geq 2$和$|\bar{A}| \geq 2$。

我们必须声明：在算法的连接过程中，下面三种情形之一将会在某一步，如第r步发生（第r步意谓着第r个顶点被完全连接的那一步）。

情形1　所有的非零余度至少为k。

情形2　所有的非零余度至少为$k-1$，并且至少存在一条由步骤1，…，r所构造的边在$\langle A, \bar{A} \rangle$中。

情形3　所有的非零余度至少为$k-2$，并且至少存在两条由步骤1，2，…，r所构造的边在$\langle A, \bar{A} \rangle$中。

所有这三种情形都意谓着：通过归纳，割集$\langle A, \bar{A} \rangle$中至少有$k$条边。

我们证明上述声明如下：

令v_i是第i步被连接的顶点。不失一般性，我们假定v_1在A_1中。我们现在证明：在连接过程的某一步中，如果情形1和情形2不会发生的话，则情形3一定会发生。

在第一步中，v_1是被完全连接的，那么

a.因为情形1不发生，所以最小的非零余度为$k-1$；

b.因为情形2不发生，所以当v_1被连接后，$\bar{A}$中没有一个顶点的度减1。

因此，v_2（第二步要被连接的顶点）必须在A中。（$\bar{A}$中的所有顶点必须至少具有度k）。现在如果v_2被连接了，且情形1不发生以及没有边连接A和$\bar{A}$，那么我们仍将有前面的（a）和（b）。因此，下一个被连接的顶点仍将在A中。

因为在连接过程的每一步中的余度都是减少的，所以早晚必然存在着一个对应于A中v_r的r，使得v_r被连接后，一个边将连接A和$\bar{A}$。如果情形2不发生，那么A中尚未完全被连接的顶点的非零余度之一变为$k-2$。这意谓着：通过连接过程，v_r必须与$\bar{A}$中的每个顶点相连接，因为$\bar{A}$中顶点都具有余度等于k，并且我们连接了v_r和最大余度的顶点。又因为$|\bar{A}| \geq 2$，所以在这一步情形3发生。□

文献[8.6]也建立了一个图序列是简单k-顶点-连通图的充要条件。

8.4　Menger定理

本节将介绍图论中的一个经典结论，即Menger定理[8.11]。这个定理有助于建立图的连通度与图中两个不同顶点间顶点-不相接路径数目的关系。

定理8.9　（**MENGER**）　图中使得两个非邻接顶点s和t不连通所移去顶点的最小数

目，等于图中顶点-不相接的$s-t$路径的最大数目。 □

这一定理的证明在15.7节中给出。

定理8.10 一个简单图$G=(V, E)$（$|V|\geqslant k+1$）是k-连通的充要条件为：在G中任两个顶点s和t之间存在着k条顶点-不相接的$s-t$路径。

证明

很显然，$k=1$时定理成立。所以对于$k\geqslant 2$，需要证明定理。

必要性 如果s和t不邻接，那么定理的必要性可由定理8.9得出。

假定s和t是邻接的，并且G中至多存在$k-1$个顶点-不相接的$s-t$路径。令$e=(s, t)$。现在考虑图$G'=G-e$。因为G中至多有$k-1$顶点-不相接的$s-t$路径，所以G'中不可能有多于$k-2$个顶点-不相接的$s-t$路径，这样，就存在着一个顶点集$A\subseteq V-\{s, t\}$，而且$|A|\leqslant k-2$，若移去这个顶点集就使得G'中的s和t不相接，因此

$$|V-A|=|V|-|A|\geqslant k+1-(k-2)=3$$

并且在$V-A$中存在着一个不同于s和t的顶点u。

现在我们证明，在G'中存在着一条$s—u$路径，它不含A中的任何顶点。很显然，如果s和u是邻接的，则它是成立的。如果s和u不邻接，则G中存在k个顶点-不相接的$s—u$路径，因而G'中存在着$k-1$个顶点-不相接的$s-u$路径。因为$|A|\leqslant u-2$，所以这$k-1$条路径中至少有一条不含A中的任何顶点。

用同样的方法，我们可以证明，G'中存在着一条不含有A中的任何顶点的$u-t$路径。

这样，G'中就存在着一条不含A中的任何顶点的$s-t$路径。可是，这与A是G'中一个$s-t$非连通集相矛盾。

因此，必要性得证。

充分性 因为在G中任何两个不同顶点之间存在着k个顶点-不相接的路径，所以G是连通的。进而因为G中没有平行边，所以有不止一条这样的路径长度为1，剩余的$k-1$条边的并必须至少含有除s和t之外的$k-1$个不同顶点。因此

$$|V|\geqslant(k-1)+2>k$$

假定在G中存在着一个非连通集A，且$|A|<k$。然后考虑对于顶点集$V-A$的G的子图G'。这个图至少含有两个不同片。如果我们从G'的任意两个不同片中各取两个顶点s和t，那么在G中至多存在$|A|<k$个顶点-不相接的$s-t$路径，这就与任意两个顶点可由图G中k个顶点-不相接的路径相连接相矛盾。

因此，充分性得证。 □

上面的结论是由文献[8.12]提出的。因为它仅仅是定理8.9的一个变形，所以我们也称它为Menger定理。

下面考虑两种k-连通图的特殊类型，即2-连通图和3-连通图。2-连通图有几个等效的特征，其中有些我们已编入第一章的习题。

Tutte给出了一个3-连通图的特征。这个特征是以一个称为轮的特殊类型的图给出的。

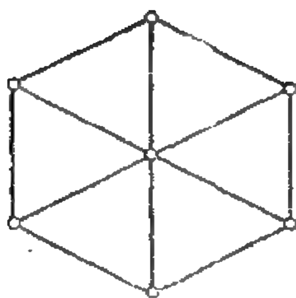


图 8.6 轮形图W_7

考虑一个长度为n的回路C。如果我们添加一个新的顶点，并且将它与C中所有顶点都连接，那么就可以得到一个($n+1$)-轮形图W_{n+1}。例如，W_7如图8-6所示。

文献[8.13]的3-连通图特征如下面的定理所述。

定理8.11 一个简单图G是3-连通的，当且仅当G是一个轮形图，或者G可从一个轮形图通过一系列如下运算而得到：

1. 添加一条新的边。
2. 用两个度≥3的邻接顶点v'和v''替换一个度≥4的顶点v，以使得每一个原来与v相邻接的顶点现在恰好与v'和v''中的一个顶点相邻接。

我们用k-边-连通图的一个特征来结束这一节。这个特征类似于定理8.9，所以也称为Menger定理，虽然它是由文献[8.14]以及文献[8.15]各自单独发现的。

定理8.12 连通图G中使得两个不同的顶点s和t不相连所移去的最小边数，等于G中边-不相接的s—t路径的最大数。 □

上面定理的证明也在15.7节中给出。

8.5 匹 配

本节和本章其余各节的讨论都涉及到所谓结婚问题及其有关的问题。

一个有限的男孩集合，其中每个男孩都有几个女朋友。在什么条件下，才能够使每个男孩都与他的女友中的一人结婚？（当然，一个女孩不能同一个以上的男孩结婚！）

这个问题可用如下图论术语提出：

构造一个二分图G，使顶点x_1，x_2，…，x_n表示男孩，顶点y_1，y_2，…，y_m表示女孩。当且仅当y_j是x_i的女友，G中才出现一条边(x_i，y_j)。这样，结婚问题就等效为在G中寻找一个边的集合，以使它们当中没有两条边有公共的顶点，而且每个x_i都是这些边中的一个端顶点。

例如，假定有4个男孩b_1，b_2，b_3和b_4与4个女孩g_1，g_2，g_3和g_4，他们之间的关系如下：

$$b_1 \longrightarrow \{g_1\}$$
$$b_2 \longrightarrow \{g_2\}$$
$$b_3 \longrightarrow \{g_1,\ g_2\}$$
$$b_4 \longrightarrow \{g_3,\ g_4\}$$

表示这种关系的二分图，如图8.7所示。很容易验证，所有4个男孩都同他的女友结婚是不可能的。可是，我们能够使3个男孩找到配偶而不违反结婚问题的要求。例如，两组这样合适的对是

1. (b_1, g_1)，　(b_2, g_2)，　(b_4, g_3)
2. (b_1, g_1)，　(b_2, g_2)，　(b_4, g_4)

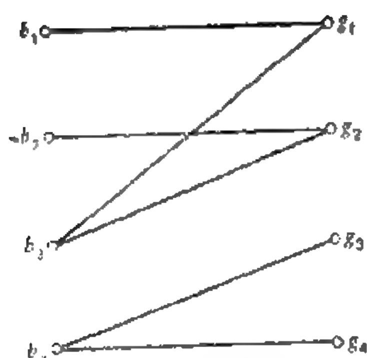


图 8.7

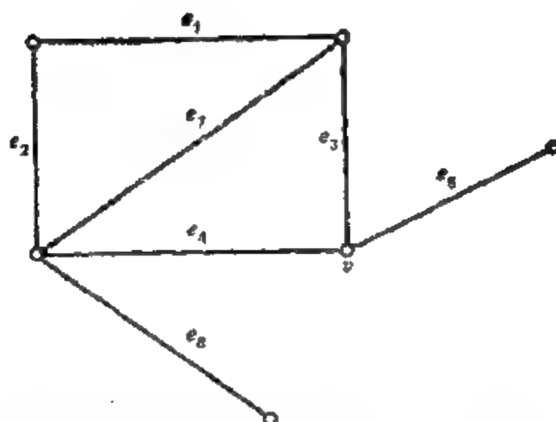


图 8.8

上面的讨论形成了图中匹配的定义。

如果图中的两条边没有任何公共顶点，则称它们是独立的。如果边e_1，e_2，…中的每两条边都没有公共顶点，那么也称它们是独立的。

图中的匹配是一个独立边的集合。例如，在图8.8中，$\{e_1, e_4\}$就是一个匹配。

很显然，极大匹配是独立边的极大集合。因此在图8.8中，$\{e_1, e_4\}$是一个极大匹配，而$\{e_5, e_8\}$就不是一个极大匹配。

具有最大边数的匹配称为最大匹配。在图8.8中，集合$\{e_1, e_5, e_8\}$是最大匹配。G的最大匹配中的边数称为G的匹配数，记为$\alpha_1(G)$。

如果一个顶点在M中是边的一个端顶点，则称这个顶点在匹配M中是饱和的。例如，在图8.8中，顶点v在匹配$\{e_1, e_5, e_8\}$中是饱和的。

下面将一个具有二划分(X, Y)的二分图$G=(V, E)$表示成一个三元素组(X, Y, E)。

我们说在二分图(X, Y, E)中，如果存在一个匹配M，并使得X中的每个顶点在M中是饱和的，那么集合X和Y是相匹配。然而匹配M称为X到Y的一个完全匹配。

借助于这个新的术语，结婚问题等效地变成在一个二分图(X, Y, E)中去寻找一个X到Y的完全匹配的充要条件。在下节中，我们将给出二分图中关于匹配的几个结论，另外提供结婚问题的解答。

8.6 二分图中的匹配

考虑一个二分图$G=(X, Y, E)$，令S是X的一个任意子集，并用$\Gamma(S)$表示与S中顶点相邻接的顶点集。假定$|S|>|\Gamma(S)|$，则显然不存在从S到$\Gamma(S)$的完全匹配，G中的任何匹配M至多包含$|\Gamma(S)|$个S中的顶点。因此

$$|M|\leqslant|X|-(|S|-|\Gamma(S)|) \qquad (8.10)$$

因为式(8.10)满足X的任何子集S，所以对于任何匹配M都能总结出

$$|M|\leqslant|X|-\max_{S\subseteq X}(|S|-|\Gamma(S)|) \qquad (8.11)$$

现在我们定义X的任意子图S的差$\sigma(S)$和G的差$\sigma(G)$如下：

$$\sigma(S)=|S|-|\Gamma(S)| \qquad (8.12)$$

和

$$\sigma(G)=\max_{S\subseteq X}(|S|-|\Gamma(S)|) \qquad (8.13)$$

利用（8.13）式，可以将（8.11）式重新写为

$$|M|\leqslant|X|-\delta(G) \qquad (8.14)$$

本节的主要结论是，在一个二分图$G=(X, Y, E)$中存在着一个具有$|X|-\sigma(G)$条边的匹配。换言之，我们要证明，G中的一个最大匹配的边数等于$|X|-\sigma(G)$。此结论将分两步来证明，我们首先证明$\sigma(G)=0$的情形，然后再证明$\sigma(G)>0$的情形。（注意，$\sigma(G)\geqslant 0$，因为对于空集$\varnothing$，$\sigma(\varnothing)=0$。）

注意，如果对于X的每个子集S

$$|S|\leqslant|\Gamma(S)|$$

那么情况$\sigma(G)=0$就将发生，如果对于X的某个子集S

$$|S|>|\Gamma(S)|$$

那么$\sigma(G)>0$就将发生。

下面的结论是由文献[8.16]提出的。我们的证明是基于文献[8.17]的。

定理8.13(Hall) 在一个二分图$G=(X, Y, E)$中，存在着一个从X到Y的完全匹配，当且仅当对于任意$S\subseteq X$

$$|S|\leqslant|\Gamma(S)|$$

证明

必要性 由（8.14）式显而易见，X到Y的完全匹配需要满足$\sigma(G)=0$，换言之，

$$|S|\leqslant|\Gamma(S)|,\quad S\subseteq X$$

充分性 对X中的顶点数$|X|$进行归纳来证明。

如果$|X|=1$，那么显然存在一个从X到Y的完全匹配，因为X中仅有一个顶点应该同Y中的至少一个顶点相邻接。

作为归纳假设，假定对于$|X|\leqslant m-1$的任何二分图证明的充分性都成立。然后考虑$|X|=m$的二分图$G=(X, Y, E)$。对于X的每一个子集S，设$S\leqslant|\Gamma(S)|$。现在我们通过检验如下情况来证明存在一个从X到Y的完全匹配。

情形1 对于每一个X的非空真子集S，$|S|<|\Gamma(S)|$。

选择任意一条边(x_0, y_0)，$x_0\in X$且$y_0\in Y$。令G'为从图G中移去x_0和y_0以及与它们相关联的边后所得到的图。因为$X-\{x_0\}$的每个子集S中的顶点与Y中多于$|S|$个顶点相邻接，所以它们一定与$Y-\{y_0\}$中的$|S|$个或者更多个顶点相邻接。因此，由归纳假设，这里存在着一个从$X-\{x_0\}$到$Y-\{y_0\}$的完全匹配，这个匹配与边(x_0, y_0)一起就组成了从X到Y的完全匹配。

情形2 若存在一个X的非空真子集S_0，使得$|S_0|=|\Gamma(S_0)|$。

设G'为一个包含了集合S_0和$\Gamma(S_0)$中顶点以及连接这些顶点间边的G的子图。

设G''为一个包含了集合$X-S_0$和$Y-\Gamma(S_0)$中顶点，以及连接这些顶点间边的G的

子图。

我们要证明，在G'中存在着一个从S_0到$\Gamma(S_0)$的完全匹配，在G''中存在着一个从$X-S_0$到$Y-\Gamma(S_0)$的匹配。这两个匹配合起来就组成了从X到Y的匹配。

首先考虑图G'。对于S_0的任何子集S，设$\Gamma'(S)$表示$\Gamma(S_0)$中与S中顶点相邻接的顶点。

因为对于S_0的每个子图S，

$$|S|\leqslant|\Gamma(S)|=|\Gamma'(S)|$$

所以显而易见

$$\Gamma'(S)=\Gamma(S),\quad S\subseteq S_0$$

由归纳假设可以看出，这是一个从S_0到$\Gamma(S_0)$的完全匹配。

下面再考虑图G''。对于$X-S_0$的任何子图，设$\Gamma(S)$表示$Y-\Gamma(S_0)$中与S中顶点相邻接的顶点集合。对于$S\subseteq X-S_0$，

$$\begin{aligned}|S\cup S_0|&=|S|+|S_0|\\&\leqslant|\Gamma(S\cup S_0)|\\&=|F''(S)|+|\Gamma(S_0)|\end{aligned}$$

因为$|S_0|=|\Gamma(S_0)|$，所以从上式得到

$$|S|\leqslant|\Gamma''(S)|,\quad S\subseteq X-S$$

这样再次由归纳假设可知，这里存在着一个从$X-S_0$到$Y-\Gamma(S_0)$的完全匹配。

于是充分性得证。 □

正如我们前面所观察到的，定理8.13提供了一个结婚问题的解答，下面叙述这个解答。

定理8.14（Hall） 结婚问题解答的充要条件是任一k个男孩的集合都至少有k个女友，$1\leqslant k\leqslant m$，这里m是男孩数。 □

定理8.13仅仅是把定理8.14转变为图论术语。

下面我们证明，当$\sigma(G)>0$时，最大匹配中的边数等于$|X|-\sigma(G)$。为此，我们需要下面两个引理。

引理8.1 设S_1和S_2是X的任何两个子集，则

$$\sigma(S_1\cup S_2)+\sigma(S_1\cap S_2)\geqslant\sigma(S_1)+\sigma(S_2)$$

证明

证实

$$|S_1\cup S_2|+|S_1\cap S_2|=|S_1|+|S_2| \tag{8.15}$$

仅仅是个简单的练习。

因为

$$|\Gamma(S_1\cup S_2)|=|\Gamma(S_1)\cup\Gamma(S_2)|$$

以及

$$|\Gamma(S_1\cup S_2)|\leqslant|\Gamma(S_1)\cap\Gamma(S_2)|$$

所以得到

$$|\Gamma(S_1 \cup S_2)| + |\Gamma(S_1 \cap S_2)| \leqslant |\Gamma(S_1) \cup \Gamma(S_2)| + |\Gamma(S_1) \cap \Gamma(S_2)|$$
$$= |\Gamma(S_1)| + |\Gamma(S_2)| \qquad (8.16)$$

现在将（8.15）式减去（8.16）式，则得

$$\sigma(S_1 \cup S_2) + \sigma(S_1 \cap S_2) \geqslant |S_1| + |S_2| - |\Gamma(S_1)| - |\Gamma(S_2)|$$
$$\geqslant^{*} \sigma(S_1) + \sigma(S_2)$$

□

引理8.2 设S_1和S_2为X的两个子集，并使得$\sigma(S_1)=\sigma(S_2)=\sigma(G)$，那么

$$\sigma(S_1 \cup S_2) = \sigma(S_1 \cap S_2) = \sigma(G)$$

证明

由引理8.1，

$$\sigma(S_1 \cup S_2) + \sigma(S_1 \cap S_2) \geqslant \sigma(S_1) + \sigma(S_2) = 2\sigma(G)$$

因为不论$\sigma(S_1 \cup S_2)$或$\sigma(S_1 \cap S_2)$都不能大于$\sigma(G)$，所以得到

$$\sigma(S_1 \cup S_2) = \sigma(S_1 \cap S_2) = \sigma(G)$$

□

定理8.15 在一个$\sigma(G)>0$的二分图$G=(X,Y,E)$中，最大匹配中的边数等于$|X|-\sigma(G)$。

证明

设S_1，S_2，…，S_k都是差等于$\sigma(G)$的X的子集。令

$$S_0 = S_1 \cap S_2 \cap \cdots \cap S_k$$

由引理8.2，

$$\sigma(S_0) = \sigma(G) > 0$$

这样，S_0是非空的。现在我们注意到，具有差等于$\sigma(G)$的X的每个子集都包含S_0。

考虑S_0中的任意顶点x_0。设G'为从G中移去顶点x_0以及与它相关联的所有边后所得到图。显而易见，$\sigma(G')<\sigma(G)$，因为集合$X-\{x_0\}$的子集不包含S_0。我们现在证明$\sigma(G')=\sigma(G)-1$。

考虑集合$S_0'=S_0-\{x_0\}$。我们有

$$\sigma(S_0') = |S_0'| - |\Gamma(S_0')|$$
$$= |S_0| - 1 - |\Gamma(S_0')|$$

因为$\sigma(S_0')<\sigma(G)$，所以得到

$$|S_0| - 1 - |\Gamma(S_0')| < |S_0| - |\Gamma(S_0)|$$

即

$$|\Gamma(S_0')| \geqslant |\Gamma(S_0)| \qquad (8.17)$$

另一方面，因为S_0'是S_0的一个子集，所以

$$|\Gamma(S_0')| \leqslant |G(S_0)| \qquad (8.18)$$

联立（8.17）式和（8.18）式，则得

$$|\Gamma(S_0')| = |\Gamma(S_0)|$$

因此

---

* 原文为"="号，应改为"≥"号。——译者注

$$\sigma(S_0')=|S_0|-1-|\Gamma(S_0)|=\sigma(G)-1$$

因为$\sigma(G')\leqslant\sigma(G)$，所以得到$\sigma(G')=\sigma(G)-1$。

如果我们重复上面的论证，直到从X中移去适当的$\sigma(G)$个顶点的集合以及关联在这些顶点上的边，那么我们就可得到一个G的子图，其差等于零。由定理8.13，在这个子图中存在着一个完全匹配。这是一个G中含有$X-\sigma(G)$条边的匹配。由（8.14）式显而易见，这样的匹配是G的一个最大匹配。 □

我们可以将定理8.13与定理8.15合为一个定理，因为它很重要，所以将在下面介绍这个定理，并请参阅文献[8.18]和[8.19]。

定理8.16（König） 在二分图$G=(X,Y,E)$的一个最大匹配中，它的边数等于$|X|-\sigma(G)$，这里$\sigma(G)$是G的差。 □

推论8.16.1 在非空二分图$G(X,Y,E)$中，如果

$$\min_{x\in X}\{d(x)\}\geqslant\max_{y\in Y}\{d(y)\}$$

那么就存在着一个X到Y的完全匹配。

证明

令

$$\min_{x\in X}\{(d(x)\}=d_1$$

以及

$$\max_{y\in Y}\{d(y)\}=d_2$$

考虑X的任意子集A。设E_1为与A中顶点相关联的边集，E_2为与$\Gamma(A)$中顶点相关联的边集合。那么则有

$$|E_1|\geqslant|A|\,d_1$$

和

$$|E_2|\leqslant|\Gamma(A)|\,d_2$$

因为$E_1\subseteq E_2$，所以有

$$|\Gamma(A)|\,d_2\geqslant|E_2|\geqslant|E_1|\geqslant|A|\,d_1$$

因此

$$|\Gamma(A)|\geqslant|A|,\ A\subseteq X$$

这样，由Hall定理可知，这是一个从X到Y的完全匹配。 □

下面我们考虑两个与Hall定理有关的结论。

第一个结论来自于横截（transversal）理论。

设M为一个非零有限集合，且$S=\{S_1,S_2,\cdots,S_r\}$（不必非有区别）为M的非空子集的一个族，那么S的一个横截（或不同代表组成的系统）就是一个M的r个不同元素的集合，其中每个元素来自于每个集合S_i。

例如，如果$M=\{1,2,3,4,5,6\}$和$S_1=\{1,3,4\}$，$S_2=\{1,3,4\}$，$S_3=\{1,2,5\}$以及$S_4=\{5,6\}$，那么$\{1,3,2,6\}$就是族$S=\{S_1,S_2,S_3,S_4\}$的一个横截。另一方面，如果$S_1=S_2=\{1,3\}$，$S_3=\{3,4,\}$以及$S_4=\{1,4\}$，那么对于族$\{S_1,S_2,S_3,S_4\}$来说就不存在一个横截。

现在提出如下的问题：

一个集合的子集的族具有一个横截的充要条件是什么？

假定我们构造一个二分图$G=(X, Y, E)$，以使得

1. X中的顶点x_i对应于S中的集合S_i；

2. Y中的顶点y_i对应于M中元素i；

3. 边$(x_i, y_j)\subset E$，当且仅当$j\in S_i$。

现在我们可以看出，上面所提出的问题等效为在按上述方法所构造的一个二分图中，寻找一个从X到Y的完全匹配。这样，我们就可得到下面的定理，它就是Hall定理用横截理论的语言的简单重述。

定理8.17 设M为一个非空集合且$S=\{S_1, S_2, \cdots\cdots, S_r\}$为$M$子集的一个族，那么$S$就具有一条横截，当且仅当任何$k(1\leqslant k\leqslant r)$个子集$S_i$的并中都含有$M$中的$k$个元素。 □

上面定理是由Rado用纯组合学而不是用图论概念给予证明的。这个证明非常精辟，可在文献[8.20]中找到。

下面的结论是关于其元素只为"0"和"1"的矩阵。这样的矩阵称为(0,1)-矩阵。矩阵中的线在下面就是指矩阵的行或列。

定理8.18 (König和Egevary)在一个(0,1)-矩阵中，含有都是1的线的最小数等于在M的同一线中没有两个1的最大数。 □

已知一个$m\times n$阶的(0,1)-矩阵M。假定我们构造一个二分图$G=(X, Y, E)$，并使得

1. X中的顶点$x_1, x_2, \cdots\cdots, x_m$对应于$M$中的$m$行；

2. Y中的顶点$y_1, y_2, \cdots\cdots, y_n$对应于$M$中的$n$列；

3. 如果M中的(i, j)元素等于1，那么引入线$(x_i, y_j)\in E$。

如果我们现在考虑一个顶点能覆盖所有与其相关联的边，那么König-Egervary定理可以复述如下：

在一个二分图中，覆盖所有边的顶点的最小数等于图的任何最大匹配中的边数。

在第九章(定理9.2)中，我们将证明上面形式的König-Egervary定理。

8.7 一般图中的匹配

本节将导出有关一般图中匹配的几个结论。

考虑一个图$G=(V,E)$和G中的一个匹配。G中的一个交变链是一个轨迹，它的边交替地出现在M中和$(E-M)$中。例如$e_1, e_2, e_3, e_4, e_7, e_8$就是一个关于图8.9中匹配$M=\{e_2, e_4, e_8\}$的一个交变链。交变链中属于$M$的边称为暗边，属于$E-M$中的边称为亮边，因此，在上面所考虑的交变链中，$e_1, e_3, e_7$是亮边，而$e_2, e_4, e_8$是暗边。

定理8.19 设M_1和M_2是简单图$G=(V, E)$中的两个匹配。并设$G'=(V', E')$为由边集

$$M_1\oplus M_2=(M_1-M_2)\cup(M_2-M_1)$$

所诱导的G的子图。那么G'的每一片都具有如下形式之一：

1. 偶数长度回路，其边交替出现在M_1和M_2中；

2. 路径，其边交替出现在M_1和M_2中，且其端顶点在两个匹配之一中未饱和。

证明

考虑任意顶点$v \in V'$。

情形1　$v \in V(M_1-M_2)$，和$v \notin V(M_2-M_1)$，这里$V(M_i-M_j)$表示在M_i-M_j中边的顶点集。

在这种情况下，v是M_1-M_2中的一条边的端点。因为M_1是一个匹配，所以M_1-M_2中没有其他的边与v相关联。进而因为$v \notin V$相关联(M_2-M_1)，所以M_2-M_1中的边都不与v相关联。因此，在这种情况下，G'中v的度等于1。

情形2　$v \in V(M_1-M_2)$和$v \in V(M_2-M_1)$。

在这种情况下，M_1-M_2中有一条边与v相关联，且M_2-M_1中有一条边与v相关联。因此，v的度为2。

因为所考虑的两种情况已穷举尽了，所以G'中的最大度为2。因此，连通片将是定理中所描述的两种形式之一。 □

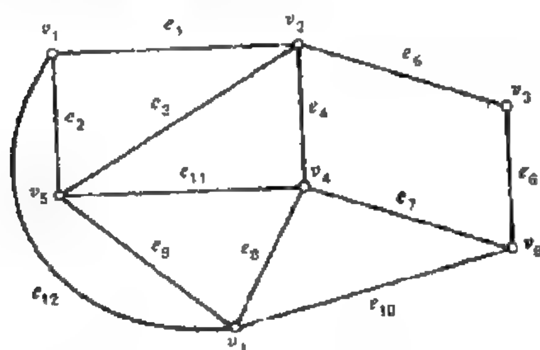


图 8.9

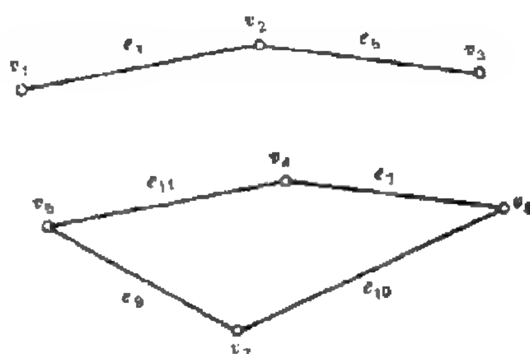


图 8.10

例如，考虑图8.9的两个匹配$M_1=\{e_5, e_7, e_9\}$和$M_2=\{e_1, e_{10}, e_{11}\}$。则

$$M_1 \oplus M_2=\{e_1, e_5, e_7, e_9, e_{10}, e_{11}\}$$

图G'如图8.10所示，还可以看出，G'的片就是定理8.19中的两种形式。

在下面定理中，我们将给出文献[8.21]关于交变链的最大匹配特征。

定理8.20（BERGE）　一个匹配M是最大的，当且仅当在任意两个不饱和的顶点之间不存在交变链。

证明

必要性　假定在两个未饱和顶点之间存在着一个交变链P。那么用链中亮边置换暗边将得到一个匹配M_1，且

$$|M_1|=|M|+1$$

注意$M_1=(M-P)\cup(P-M)$。

例如，在图8.9中，考虑匹配$M=\{e_2, e_4\}$。在未饱和顶点v_3和v_7之间存在着一个交变链e_6，e_4，e_{11}，e_2和e_{12}。如果我们在M中用亮边e_6，e_{11}和e_{12}置换暗边e_4和e_2，那么则

得到匹配$\{e_3, e_{11}, e_{12}\}$，而它比M多一条边。

充分性　假定M满足定理条件。设M'是一个最大匹配。那么由定理的必要性可知，M'也满足定理条件，即在M'中的任意两个未饱和顶点之间不存在交变链。我们现在证明了$|M| = |M'|$，由此定理的充分性得证。

因为$M = (M \cap M') \cup (M - M')$以及$M' = (M \cap M') \cup (M' - M)$，所以显然有$|M| = |M'|$，当且仅当$|M - M'| = |M' - M|$。

设G'是由边集$M \oplus M' = (M - M') \cup (M' - M)$所诱导的图。

首先考虑G'中的回路。由定理8.19得知，这样回路的长度为偶数，而且这个回路中的边交替出现在$M - M'$和$M' - M$中。因此，G'中的每个回路在$M - M'$中的边数和$M' - M$中的边数是相等的。

其次考虑G'中组成路径的一个片。再次由定理8.19得知，这个路径中的边交替地出现在$M - M'$和$M' - M$中，进而这个路径中的端点在M或M'中是未饱和的。假定路径的长度为奇数，那么路径的端点在同样的匹配中都是未饱和的。这就意谓着，对于这两个匹配中的一个，在两个未饱和顶点之间存在着一个交变链。但是因为M和M'都满足定理的条件，所以这是相矛盾的。因此在G'中组成路径的每个片都具有偶数条边，而且$M - M'$和$M' - M$中的边数是相同的。

这样，G'的每个片在$M - M'$和$M' - M$中都具有相同的边数。因为G中边数构造了集合$(M - M') \cup (M' - M)$，所以得到

$$|M - M'| = |M' - M|$$

因此$|M| = |M'|$。 □

已知图G中的一个匹配M。设P是任意两个在M中未饱和顶点之间的一个交变链。正如我们前面所看到的，$M \oplus P$是一个比M多一条边的匹配，因此，路径P称为相对于M的一个增广路径。

下面我们用交变链的理论来证明关于二分图中的两个有趣的结论。

考虑一个具有最大度Δ的二分图$G = (X, Y, E)$。设X_1表示X中度为Δ的所有顶点的集合。如果G'是二分图$(X_1, \Gamma(X_1), E')$，这里E'是连接X_1和$\Gamma(X_1)$的边的集合，那么由推论8.16.1可以看出，G'中存在着一个从X_1到$\Gamma(X_1)$的完全匹配。这样的匹配显然饱和了X_1中的所有顶点。因此，在G中存在着一个饱和了X中度为Δ的所有顶点的匹配。同样地，在G中存在着一个饱和了Y中度为Δ的所有顶点的匹配。现在提出的问题是，在一个二分图中，是否存在着一个饱和了X和Y中所有最大度的顶点的匹配。为了解答这个问题，我们需要如下的结论。该结论是由文献[8.22]提出的

定理8.21（MENDELSOHN和DULMAGE）　设$G = (X, Y, E)$为一个二分图，并设M_i是一个从$X_i \subset X$到$Y_i \subset Y$ $(i = 1, 2,)$的匹配，那么就存在着一个饱和了X_1和Y_2的匹配$M' \subseteq M_1 \subset M_2$。

证明

考虑二分图$G' = (X_1 \cup X_2, Y_1 \cup Y_2, M_1 \cup M_2)$。这个图的每个顶点的度为1或2，因此这个图的每个片或是一条路径，或是一个回路，它的边交替地出现在M_1和M_2中（见

定理8.19的证明)。

G'中的每个顶点$y\in Y_2-Y_1$的度为1，所以它在一条路径为P_y的连通片中，这条路径是从y到一个顶点$x\in X_2-X_1$或者到另一个顶点$z\in Y_1-Y_2$的。在前一种情况下，P_y的最后一条边在M_2中，所以$M_1\oplus P_y$是一个从$X_1\cup\{x\}$到$Y_1\cup\{y\}$的匹配。在后一种情况，P_y的最后一条边在M_1中，所以$M_1\oplus P_y$是一个从X_1到$(Y_1-z)\cup\{y\}$的匹配。在任何一种情况下，$M_1\oplus P_y$都饱和了$Y_1\cap Y_2$。这样，$M_1\oplus P_y$就饱和了$y\in Y_2-Y_1$以及X_1和$Y_1\cap Y_2$中的所有顶点。

如果我们设

$$P=\bigcup_{y\in Y_2-Y_1}P_y$$

那么就可以看到$M_1\oplus P$是一个饱和了X_1和Y_2的匹配，这就是所需要的匹配$M'\subseteq M_1\cup M_2$。 □

定理8.22 在一个二分图中，存在着一个饱和了所有最大度顶点的匹配。

证明

考虑二分图$G=(X, Y, E)$。设$X'\subseteq X$以及$Y'\subseteq Y$包含了G中所有最大度的顶点。这正如我们前面所看到的，其中存在着一个饱和了X'中所有顶点的匹配M_1和一个饱和了Y'所有顶点的匹配M_2。所以由定理8.21可知，这里存在着一个匹配$M'\subseteq M_1\cup M_2$，它饱和了X'和Y'中的所有顶点。这就是一个所需饱和G中所有最大度顶点的匹配。 □

推论8.22.1 具有最大顶点度Δ的二分图的边集可以被划分为Δ个匹配。

证明

考虑具有最大顶点度Δ的二分图$G=(X,Y,E)$。由定理8.22可知，这里存在着一个饱和了所有度为Δ的顶点的匹配M'。因此二分图$G'=(X, Y, E-M_1)$具有最大度$\Delta-1$，这个图又含有一个饱和了所有度为$\Delta-1$的顶点的匹配M_2。重复这一次程，我们可以构造出一个不相交的匹配序列$M_1, M_2, \cdots, M_\Delta$，这就形成了$E$的一个划分。

定理8.21和定理8.22以及推论8.22.1的应用将在15.6节讨论。 □

现在我们回到讨论一般图的匹配。

一个饱和了图G中所有顶点的匹配称为G的完美匹配。

下面我们利用文献[8.23]关于一个图中完美匹配存在条件的定理来结束本节。这里的证明出自于文献[8.24]。

如果图的一个片中的顶点为奇数，那么这个片是奇的。否则它就是偶的。如果S是图G顶点的一个子集，那么我们就用$p_0(S)$来表示$G-S$的奇片数。

定理8.23 (TUTTE) 图$G=(V, E)$具有一个理想匹配，当且仅当

$$p_0(S)\leqslant|S|,\quad 对于所有S\subset V \tag{8.19}$$

证明

必要性 假定G具有一个完美匹配M。对于某些$S\subset V$，设$G_1, G_2, \cdots, G_k$为$G-S$的奇片。因为每个G_i是奇的，所以G_i中的某些顶点v_i必须在M下同S的某些顶点v_j相匹配。这样S至少有k个顶点，因而

$$p_0(S)\leqslant|S|$$

充分性　首先注意到，如果图G满足(8.19)式，那么选择$S=\varnothing$就得到$p_0(\varnothing)\leqslant 0$。所以在这种情况下，$G$中将不存在奇片，换言之，$G$具有偶数个顶点。

充分性的证明是通过对l归纳进行的，这里$2l=|V|$。我们还利用Hall定理（定理8.13）和简单的事实

$$p_0(S)=|S| \quad (\text{模 } 2) \tag{8.20}$$

$l=1$的情况是无关紧要的。假定结论对于所有顶点数小于$2l$的图都是正确。然后考虑一个具有$2l$个顶点并且满足（8.19）式的图G。我们分两种情况讨论。

情形1　　假定$p_0(S)<|S|$，对所有的S，$2\leqslant|S|<2l$。

考虑图中的任意边$e=(a, b)$。设$A=\{a, b\}$以及$G_A=G-A$。对于G_A中顶点的任意子集T，令$p_0'(T)$表示G_A-T中的奇片数。于是$p_0'(T)\leqslant|T|$，因为如果$p_0'(T)>|T|$，那么就得到了$p_0(T\cup A)=p_0'(T)>|T|=|T\cup A|-2$，这与(8.19)式相矛盾。因此由归纳假设可知，$G_A$具有一个完美匹配，即$G$具有一个完美匹配。

情况2　　令存在着一个集合S，它使得$p_0(S)=|S|\geqslant 2$，并假设S是一个最大的这样的集合。

首先观察到，在$G-S$中不存在偶片。如果$G-S$中存在任何偶片的话，那么就从这个片中移去顶点v并把它加到S上。这样就必须至少增加一个奇片，因此$p_0(S\cup v)\geqslant p_0(S)+1=|S|+1$。而条件(8.19)要求$p_0(S\cup v)\leqslant|S|-1$。所以$p_0(S\cup v)=|S|+1$，但这与$S$的最大性相矛盾。因此，$G-S$中不存在偶片。

设$|S|=s$，并设G_1，G_2，…，G_s为$G-S$中的奇片。现在来证明：我们能够从这些奇片的每个片中拿出一个顶点，并使得它与S中的顶点相匹配。如果这样不可能的话，那么由Hall定理可知，在G中就存在k个仅与S中$h<k$个顶点相连接的奇片。但是如果用T表示这h个顶点的集合，那么我们就有

$$p_0(T)\geqslant k>h=|T|$$

但这与(8.19)式相矛盾。这样我们就能够从每个G_i中拿出一个顶点$v_i(1\leqslant i\leqslant s)$，使得它与$S$的顶点相匹配。

现在每个$G_i'=G_i-v_i$都具有偶数个顶点。如果我们证明了每个G_i'都具有一个完美匹配，那么本证明就以此结束。

如果G_i'含有一个使得$p_0^i(R)>|R|$的顶点集R，这里$p_0^i(R)$是$G_i'-R$中的奇片数，那么由(8.20)式可知，$p_0^i(R)\geqslant|R|+2$。因此

$$\begin{aligned} p_0(R\cup S\cup\{v_i\}) &= p_0^i(R)+p_0(S)-1 \\ &\geqslant |R|+|S|+1 \\ &= |R\cup S\cup\{v_i\}| \end{aligned} \tag{8.21}$$

但是条件(8.19)要求

$$p_0(R\cup S\cup\{v_i\})\leqslant|R\cup S\cup\{v_i\}|$$

所以

$$p_0(R\cup S\cup\{v_i\})=|R\cup S\cup\{v_i\}|$$

这与S的最大性相矛盾。因此由归纳假设得知，G_i'具有一个完美匹配。　□

文献[8.25]利用定理8.20给出了上面定理的另一种证明。

8.8 进一步阅读

文献[8.26]和[8.8]是值得推荐为进一步阅读关于连通度和匹配的两本文献，文献[8.26]还对有向和无向图情况下度序列的实现作了详细的讨论。文献[8.8]给出了 Menger 定理的历史说明以及它的几种变形。

一个通讯网络可以模拟为一个图。易损性概念产生于研究用图模拟的网络中。我们谈到的易损性是指网络对干扰的敏感度。如果移去图中某些顶点和边后，就使得图不连通了，那么我们就认为这个网络被"损坏"了。这样，网络的易损性就与网络顶点和边的连通度有关了。例如，如果网络N_1的顶点连通度小于网络N_2的顶点连通度，那么我们就认为N_1比N_2更容易损坏。

文献[8.27]有几篇关于具有特定连通度和可靠性的图设计的论文。有关这一题目的其它几篇文章包括在文献[8.28]、[8.29]以及[8.30]中。文献[8.31]也可作为关于这一题目的阅读资料。

一个图的连通度测试问题与寻找一个传输网络中的最大流量问题是密切相关的。进一步的讨论和有关的参考文献见15.7节，那里我们也证明了Menger定理。

文献[8.32]、[8.33]的文章和文献[8.34]的书都是值得推荐的、关于匹配和横截理论的更高一级的阅读资料。有关匹配理论的应用（特别是最优分配问题和时间表问题）和有关算法将在15.4到15.6节中讨论。

8.9 习　　题

8.1 设$d_1 \leqslant d_2 \leqslant \cdots \leqslant d_n$表示简单图$G$各个顶点的度。证明：如果

(a) $d(v_r) \geqslant r+k-1$，$r \leqslant \frac{n-k}{2}$；

(b) $d(v_{n-k+1}) \geqslant \frac{n+k-2}{2}$

那么图G就是k-连通的，$k<n$。

8.2 证明：对于一个简单的n-顶点图G，如果

$$\delta(G) \geqslant \frac{n+k-2}{2}$$

那么它就是k-连通的。

8.3 设$G=(V, E)$是一个简单的k-连通图。令$B=\{b_1, b_2, \cdots, b_k\}$是一个$|B|=k$的顶点集。如果$a \in V-B$，那么证明：$G$中存在着$k$条从$a$到$B$的顶点-不相接的$a-b_i$条路径。

8.4 设G是一个简单的k-连通图，且$k \geqslant 2$。证明：G中存在着一个经过任意二条边e_1和e_2以及$k-2$个顶点集的回路。

8.5 证明：$k(H_{k,n})=k'(H_{k,n})=k$。

8.6 画出一个具有7个顶点、18条边的5-连通图。

8.7 证明：一个简单图G的边可以在给定方向后，构成一个强连通图，当且仅当G是2-边-连通的。

8.8 无向图G的伴随有向图$D(G)$是一个用两条反向、且具有相同端顶点的边去置换G中的每条边e后所得到的图。证明：

(a) 在G中的路径和$D(G)$中的有向路径之间存在着一一对应的关系;

(b) $D(G)$是k-边-连通图，当且仅当G是k-边-连通图。

注：有向图G是k-边-连通的，当且仅当对于G中任意两个顶点s和t来说，至少要移去k条边才能破坏所有的s—t路径。

8.9 画出一个具有度序列（5，4，4，4，3，3，3，3，3）和最大可能的边连通度的简单图。

8.10 证明：一个非负整数序列（d_1，d_2，…，d_n）是一个树的度序列，当且仅当

$$d_i \geq 1，对于所有i$$

和

$$\sum_{i=1}^{n} d_i = 2(n-1)$$

8.11 证明：非负整数序列（d_1，d_2，…，d_n）是一个图的度序列，当且仅当$\sum_{i=1}^{n} d_i$是偶数（见文献[8.4]）。

8.12 证明：非负整数序列（d_1，d_2，…，d_n）是一个简单图的度序列，当且仅当

（a）$\sum_{i=1}^{n} d_i$是偶数;

（b）$\sum_{i=1}^{k} d_i \leq k(k-1) + \sum_{i=k+1}^{n} \min\{k, d_i\}$，对于$1 \leq k \leq n-1$

（见文献[8.7]）

8.13 证明：序列（d_1，d_2，…，d_n）（可由简单图实现）是一个简单k-连通图的度序列，当且仅当

(a) $d_i \geq k$，对于$1 \leq i \leq n$，

(b) $m - \sum_{i=1}^{k-1} d_i + \frac{(k-1)(k-2)}{2} \geq n-k$

这里m是G中的边数（见文献[8.6]）。

8.14 证明或否定：对于每个匹配M，则存在着一个最大匹配M'，并使得$M \subseteq M'$。

8.15 如果非负实方阵P的每行元素之和为1以及P的每列元素之和也为1，那么非负实方阵P就是双重随机的。一个排列矩阵就是每行或每列恰好只有一个1的（0，1）-矩阵。证明：双重随机矩阵P可以表示为

$$P = c_1P_1 + c_2P_2 + \cdots + c_kP_k$$

这里每个P_i是个排列矩阵，每个c_i是一个非负实数，且$\sum_{i=1}^{k} c_i = 1$。

8.16 设G是一个具有二划分（X，Y）的二分图。如果G有一个从X到Y的完全匹配，那么证明：G中存在一个$x_0 \in X$，并使得对于每个$y \in \Gamma(x_0)$，至少有一个最大匹配包含了边（x_0，y）。

8.17 图G的k-因子是一个G的k-正则生成子图。很显然，一个1-因子就是一个理想匹配。如果G是某些边-不相连k-因子的并，那么G就是k-可因子化的。证明：$K_{n,n}$和K_{2n}都是1-可因子化的。

8.18 (a) 证明：K_{2n+1}可以表示为n个连通2-因子的并（$n \geq 1$）。

注：一个连通2-因子是一个哈密顿回路。

(b) 证明：K_{2n}是一个1-因子和$n-1$个连通2-因子的并。

8.19 证明：连通图G是2-可因子化的，当且仅当它是偶数度的正则图。

8.20 设M和N为图G的两个边-不相接的匹配，而$|M| > |N|$。证明：G中存在着不相交的矩阵M'和N'，且$|M'| = |M| - 1$、$|N'| = |N| + 1$以及$M' \cup N' = M \cup N$。

8.21 如果$G=(V, E)$是一个二分图，且$k\geqslant\Delta$，那么证明：存在着k个G的不相交匹配$M_1, M_2, \cdots, M_k$，并使得

$$E=M_1\cup M_2\cup\cdots\cup M_k$$

而且对于$1\leqslant i\leqslant k$

$$\left\lfloor\frac{m}{k}\right\rfloor\leqslant|M_i|\leqslant\left\lceil\frac{m}{k}\right\rceil$$

这里m是G的边数。

8.22 证明：树T具有一个理想匹配，当且仅当对于T中所有顶点v，$p_0(v)=1$。这里$p_0(v)$是$T-v$的奇片数。

8.23 从Tutte定理推导出Hall定理（见文献[8.35]中的习题5.3.1）。

8.24 从Meger定理推导出Hall定理（见文献[8.36]中的定理28d）。

8.10 参考文献

8.1 F. Haray, "The Maximum Connectivity of Graph," *Proc. Nat. Acad. Sci., U.S.*, Vol. 48, 1142—1146(1962).

8.2 J. A. Bondy, "Properties of Graphs with Constraints on Degrees," *Studia Sci. Math. Hung.*, Vol. 4, 473—475(1969).

8.3 G. Chartrand, "A Graph Theoretic Approach to a Communication Problem," *SIAM J. Appl. Math.*, Vol. 14, 778—781(1966).

8.4 S. L. Hakimi, "On the Realizability of a Set of Integers as Degrees of the Vertices of a Graph," *SIAM J. Appl. Math.*, Vol. 10, 496—506(1962).

8.5 V. Havel, "A Remark on the Existence of Finite Graphs" (in Hungarian), *Časopois Pěst. Mat.*, Vol. 80, 477—480(1955).

8.6 D. L. Wang and D. J. Kleitman, "On the Existence of n-Connected Graphs with Prescribed Degrees($n>2$)," *Networks*, Vol. 3, 225—239(1973).

8.7 P. Erdös and T. Gallai, "Graphs with Prescribed Degrees of Vertices" (in Hungarian), *Mat. Lapok*, Vol. 11, 264—274(1960).

8.8 F. Harary, *Graph Theory*, Addison-Wesley, Reading, Mass., 1969.

8.9 J. Edmonds, "Existence of k-Edge-Connected Ordinary Graphs with Prescribed Degrees," *J. Res. Nat. Bur. Stand. B.*, Vol. 68, 73—74(1964).

8.10 D. L. Wang and D. J. Kleitman, "A Note on n-Edge Connectivity," *SIAM J. Appl. Math.*, Vol. 26, 313—314(1974).

8.11 K. Menger, "Zur allgemeinen Kurventheorie," *Fund. Math.*, Vol. 10, 96—115 (1927).

8.12 H. Whitney, "Congruent Graphs and the Connectivity of Graphs," *Am. J. Math.*, Vol. 54, 150—168(1932).

8.13 W. T. Tutte, "A Theory of 3-Connected Graphs," *Indag. Math.*, Vol. 23, 441—455(1961).

8.14 L. R. Ford and D. R. Fulkerson, "Maximal Flow Through a Network," *Canad. J. Math.*, Vol. 8, 399—404(1956).

8.15 P. Elias, A. Feinstein, and C. E. Shannon, "A Note on the Maximum Flow

Through a Network," *IRE Trans. Inform. Theory*, IT-2, 117—119(1956).

8.16 P. Hall, "On Representatives of Subsets," *J. London Math. Soc.*,Vol. 10, 26—30 (1935).

8.17 P. R. Halmos and H. E. Vaughan, "The Marriage Problem," *Am. J. Math.*, Vol. 72, 214—215(1950).

8.18 D. König, "Graphs and Matrices" (in Hungarian), *Mat. Fiz. Lapok*, Vol. 38, 116—119(1931).

8.19 O. Ore, "Graphs and Matching Theorems," *Duke Math. J*, Vol. 22, 625—639 (1955).

8.20 R' Rado, "Note on the Transfinite Case of Hall's Theorem on Represenlatives," *J. London Math. Soc.*, Vol. 42, 321—324(1967).

8.21 C. BerTe, "Two Theorems in Graph Theory," *Proc. Nat. Acad. Sci. U. S.*, Vol. 43, 842—844(1957).

8.22 N. S. Mendelsohn and A. L. Dulmage, "Some Generalizations of the Problem of Distinct Representatives," *Canad. J. Math.*, Vol. 10, 230—241(1958).

8.23 W. T. Tutte, "The Factorization of Linear Graphs," *J. London Math. Soc.*, Vol. 22, 107—111(1947).

8.24 I. Anderson, "Perfect Matchings of a Graph," *J. Combinatorial Theory B*, Vol. 10, 183—186(1971).

8.25 L. Lovász, "Three Short Proofs in Graph Theory," *J. Combinatorial Theory B*, Vol. 19, 111—113(1975).

8.26 C. Berge, *Graphs and Hypergraphs*, North Holland, Amsterdam, 1973.

8.27 F. T. Boesch (Ed), *Large-Scale Networks: Theory and Design*, IEEE Press, New York, 1976.

8.28 S. L. Hakimi, "An Algorithm for Construction of Least Vulnerable Communication Networks or the Graph with the Maximum Connectivity," *IEEE Trans. Circuit Theory*, Vol. CT-16,229—230(1969).

8.29 F. T. Boesch and R. E. Thomas, "On Graphs of Invulnerable Communication Nets," *IEEE Trans. Circuit Theory*, Vol. CT-17,183—192(1970).

8.30 A. T. Amin and S. L. Hakimi, "Graphs with Given Connectivity and Independence Number or Networks with Given Measures of Vulnerability and Survivability," *IEEE Trans. Circuit Theary*, Vol. CT-20,2—10(1973).

8.31 H.Frank and I. T. Frisch, *Communication, Transmission, and Transportation Nerworks*, Addison-Wesley, Reading, Mass. 1971.

8.32 L. Mirsky and H. Perfect, "Systems of Representatives," *J. Math. Anal. Applic.*, Vol. 15, 520—568(1966).

8.33 R. A. Brualdi, "Transversal Theory and Graphs," in *Studies in Graph Theory*, Part II, MAA Press, 1975, pp. 23—88.

8.34 L. Mirsky, *Transversal Theory*, Academic Press, New York, 1971.

8.35 J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, Macmillan, London, 1976.

8.36 R. J. Wilson, *Introduction to Graph Theory*, Oliver and Boyd, Edinburgh, 1972.

第九章　覆盖和着色

在前几章中，我们定义了一些和图有关的有用参数，如秩、零度、连通度、匹配数，等等。我们以前提到的秩和零度都是从研究电网络中提出来的，而连通度是从研究通讯网中提出来的。在本章，我们将研究图的另外几个有用参数-顶点独立数，顶点和边覆盖数，色指数和色数。我们开始先对顶点独立数、顶点和边覆盖数进行讨论，再把这些数和前一章定义的匹配数联系起来，并且推导Hall定理的等效表达式。然后，我们再研究色指数和色数，这些都是和一个图的顶点和边着色性质相联系的。

本章所讨论的参数都是从研究一些实际问题中提出来的。例如，时间表的安排和通讯网的研究。

9.1　独立集和顶点覆盖

考虑图$G=(V, E)$。S是V的一个子集，如果在S中任意两个顶点在G中都是不邻接的，那么S就是G的一个独立集。独立集也称为稳定集。

如果G中不存在具有$|S'|>|S|$的独立集S'，那么G的独立集S称为最大的。G的最大独立集S中的顶点数称为G的独立数（稳定数），并记为$\alpha_0(G)$。

例如，在图9.1中，集合$\{b, d\}$、$\{b, f\}$、$\{a, c\}$和$\{b, d, f\}$都是独立集。在这些集合中，$\{b, d\}$和$\{b, f\}$不是极大的独立集；$\{a, c\}$是极大的，但不是最大的；$\{b, d, f\}$是最大的。

K是V的一个子集，如果G的每条边至少有一个端点在K中，则称K为G的一个顶点覆盖。如果我们认为一个顶点覆盖了关联在其上的所有边，那么G的一个顶点覆盖就是V的一个子集，而且覆盖了G中所有边。

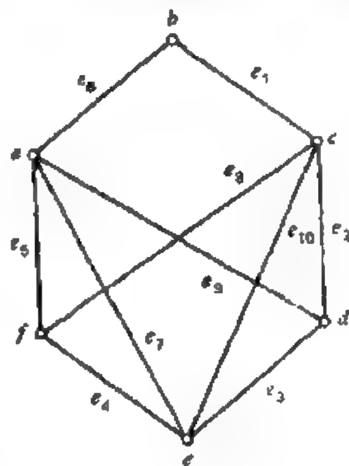


图 9.1

如果在G中不存在具有$|K'|<|K|$的一个顶点覆盖K'，那么G的顶点覆盖K称为最小的。在G的最小顶点覆盖中的顶点数称为G的顶点覆盖数，并记为$\beta_0(G)$。

例如，在图9.1中，集合$\{a, c, e, f\}$、$\{a, c, d, e\}$、$\{b, d, e, f\}$和$\{a, c, e\}$都是顶点覆盖。在这些集合中，$\{a, c, e, f\}$和$\{a, c, d, e\}$不是极小的；$\{b, d, e, f\}$是极小，也不是最小的；$\{a, c, e\}$是最小的。

只要在上下文中所考虑的图G是确定的话，那么，$\alpha_0(G)$和$\beta_0(G)$就可分别简记为α_0和β_0。曾记得：$\alpha_1(G)$是G的最大匹配中的边数。

独立集和顶点覆盖是紧密联系的二个概念，正象下面的定理给出的那样。

定理9.1 考虑图$G=(V, E)$。V的一个子集S是G的一个独立集，当且仅当V中S的补$\bar{S}$（即$\bar{S}=V-S$）是G的一个顶点覆盖。

证明

根据S的定义，S是G的一个独立集，当且仅当，G中不存在两个端点都在S中的边。换言之，S是独立集，当且仅当G中每条边至少有一个端点在$\bar{S}$中，而且$\bar{S}$是V中S的补。根据顶点覆盖的定义，则本定理得证。 □

推论9.1.1 对于n个顶点的简单图，$\alpha_0+\beta_0=n$。

证明

考虑图$G=(V, E)$的一个最大独立集S^*和一个最小顶点覆盖K^*。那么

$$|S^*|=\alpha_0$$

和

$$|K^*|=\beta_0$$

根据定理9.1，$\bar{S}^*=V-S^*$是一个顶点覆盖，$\overline{K}^*=V-K^*$是一个独立集，所以，

$$|\bar{S}^*|=|V-S^*|=n-\alpha_0\geqslant\beta_0$$

和

$$|\overline{K}|=|V-K^*|=n-\beta_0\leqslant\alpha_0$$

联立上面二个不等式，可得

$$\alpha_0+\beta_0=n$$ □

考虑图G中任一最大匹配M^*和任一最小顶点覆盖K^*。因为至少需要$|M^*|$个顶点来覆盖M^*中的边，所以顶点覆盖K^*必须至少包含$|M^*|$个顶点。因此，

$$|M^*|\leqslant|K^*| \tag{9.1}$$

通常（9.1）式中的等号是不满足的。然而只要G是一个二分图，在下面的定理中将证明$|M^*|=|K^*|$。

定理9.2 对于一个二分图，最大匹配中的边数等于最小顶点覆盖中的顶点数，即

$$\alpha_1=\beta_0。$$

证明

设M^*为二分图$G=(X, Y, E)$的最大匹配，K^*为它的最小顶点覆盖。

考虑X的任一子集A。G中每条边e不是与A中的顶点关联，就是与$X-A$中的顶点关联。进而，如果任一条边与A中的顶点关联，它也一定与$\Gamma(A)$中的顶点关联，那么$\Gamma(A)$就是与A中顶点相邻接的顶点集合。因此，集合$(X-A)\cup\Gamma(A)$是G的一个顶点覆盖，所以

$$|(X-A)|+|\Gamma(A)|\geqslant|K^*|$$

但是根据Hall定理（定理8.13）

$$\left|M^*\right|=\min_{A\subseteq X}\left\{\left|X-A\right|+\left|\Gamma(A)\right|\right\}\geqslant\left|K^*\right| \tag{9.2}$$

联立（9.1）式和（9.2）式可得

$$|M^*|=|K^*|$$ □

上述定理是由文献[9.1]给出的。

在定理8.20中，我们用交变链的概念，确立了最大匹配的特征。因为独立集是匹配的顶点模拟，所以可望在最大独立集中也存在类似的特征。这确实如此，并且在下面将推导这个特征。我们按文献[9.2]给出的方法来论述。

让我们先定义一个交变序列，即交变链的顶点模拟。

设Y为图$G=(V, E)$的独立集。与Y有关联的交变序列，是这样一个序列：

$$\sigma=\{x_1, y_1, x_2, y_2, \cdots\}$$

其中不同顶点交替属于$\overline{X}=Y=V-Y$和Y，使得下述条件得到满足：

1.$x_i\in X$，$y_i\in Y$。

2.y_i至少与集合$\{x_1, x_2, \cdots, x_i\}$中的一个顶点相邻接。

3.x_{i+1}不与集合$\{x_1, x_2, \cdots, x_i\}$中的任一顶点相邻接。

4.x_{i+1}至少与集合$\{y_1, y_2, \cdots, y_i\}$中的一个顶点相邻接。

极大的交变序列的条件是：在不破坏上述条件下，它不能增加更多的顶点。

例如，在图9-1所示图中，序列$\{f, a, b, c\}$就是与独立集$\{a, c\}$有关的极大的交变序列。

考虑一个树T。回忆得知（习题2.2）T是可二分的，即，它的顶点集V能够分成二个子集X和Y，并且使得

1.$V=X\cup Y$；

2.$X\cap Y=\varnothing$；

3.X和Y在树中都是独立集。

我们认为(X, Y)是属于T的顶点集的二划分，并且X和Y的顶点分别表示为X-顶点和Y-顶点。

作为利用交变序列来推导最大独立集特性的第一步，我们须论证：如果在一个树中，$|X|>|Y|$，那么就存在一个和Y有关的交变序列，并含有Y的全部顶点。为做到这一步，我们需要如下的引理。

引理9.1 设(X, Y)是树T顶点集合的二划分。如果$|X|=|Y|+P$，$P\geqslant 0$，那么X至少包含$P+1$个T的悬挂点。

证明

证明采用树T顶点数n的归纳法。很显然，对于$n=2$、3、4，引理成立。设对顶点数小于n（$n\geqslant 5$）的所有树，结论是正确的。

考虑具有n个顶点的树，并且$|X|=|Y|+P$。设v是T的一个悬挂点，并且设

$$T'=T-\{v\}$$

注意：T'是具有$n-1$个顶点的树。

假设v属于X-顶点。如果$P=0$，那么v是所要求的T中X-顶点。另外，根据归纳的前提，至少有P个T'的悬挂顶点是X-顶点。这P个顶点在T中也是悬挂顶点。因此除了v以外，其它顶点都是T的X-顶点。

如v是Y-顶点，也可直接证明。 □

引理9.2 设(X, Y)是树T顶点集的二划分。

1　如果$|X|=|Y|$或$|X|=|Y|+1$，那么就存在一个交变序列$\{x_1, y_1, x_2, y_2, \cdots\}$，它用到$T$的每个顶点恰好一次。

2　如果$|X|>|Y|+1$，那么就存在一个长度为奇数的最大交变序列$\{x_1, y_1, \cdots\}$，它用到Y的每个顶点恰好一次。

证明

1．对于顶点为2的树，结论正确。假设对于$2k$个顶点的任一树，结论正确。然后，考虑$2k+1$个顶点的树和$|X|=|Y|+1$。根据前一个引理，在T中存在一悬挂顶点$X_{k+1}\in X$。设$T'=T-\{X_{k+1}\}$。根据归纳的前提，在T'中有一个交变序列$\{x_1, y_1, \cdots, y_k\}$，它用到了$T'$中的所有顶点。所以序列$\{x_1, y_1, \cdots y_k, x_{k+1}\}$是$T$所要求的序列（注意，这个序列的长度为奇数）。

其次假设对于$2k+1$个顶点的任一树，结论是正确的。考虑$2k+2$个顶点的树和$|X|=|Y|$。然后，根据前一个引理，在T中存在一个悬挂顶点$y_{k+1}\in Y$。设$T'=T-\{y_{k+1}\}$。现在根据归纳的前提，存在一个交变序列$\{x_1, y_1, \cdots X_{k+1}\}$，它用到了$T'$的所有顶点，所以序列$\{x_1, y_1, \cdots, x_{k+1}, y_{k+1}\}$是$T$所要求的序列。

2　如果$|X|>|Y|+1$，从T中移去足够多的悬挂X-顶点，直到我们得到树T'，而且T'中X-顶点数比Y-顶点数大1为止。根据前一引理，这总是可能的。

现在利用这个引理的第一部分，则存在一个长度为奇数的最大交变序列，它用到了T'的所有顶点，也就和T的所有Y-顶点。

例如，在图9.2所示的树中，$\{x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4, x_5, y_5, x_6, y_6, x_7\}$是一个交变序列，它用到了$T$的所有$Y$-顶点。

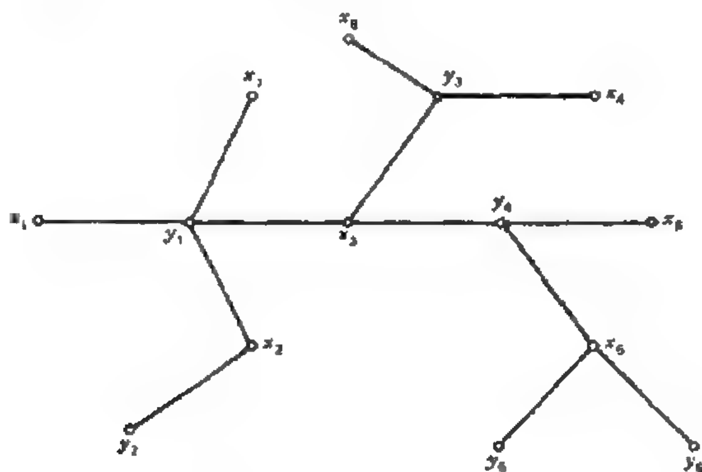


图 9.2

定理9.3　图G中的一个独立集Y是最大的，当且仅当在G中不存在与Y有联系、且长度为奇数的极大交变序列。

证明

必要性　考虑图$G=(V, E)$，设Y是G中的最大独立集，并且$X=V-Y$。

假设这里存在一个与Y有联系、且长度为奇数的极大交变序列σ，并设

$$\sigma=\sigma_1\cup\sigma_2$$

这里

$$\sigma_2=\sigma\cap Y$$

和

$$\sigma_1=\sigma\cap X$$

注意因为σ为奇长度，所以$|\sigma_1|>|\sigma_2|$，并且σ的最后一个顶点是X-顶点。然而，因为σ是极大的，所以顶点$y_j\in Y-\sigma_2$不能加入σ中，从而不破坏交变序列定义的条件（2）。

因为，不存在顶点$y_j\in Y-\sigma_2$和任一顶点$x_i\in\sigma_1$相邻接。因为σ_1和$(Y-\sigma_2)$本身都是独立集，所以$(Y-\sigma_2)\cup\sigma_1$是一个独立集。

这样，

$$|(Y-\sigma_2)\cup\sigma_1|>|(Y-\sigma_2)|+|\sigma_2|=|Y|$$

与Y是一个最大的独立集相矛盾。

充分性　设X是一个最大独立集，而Y是一个独立集，且$|Y|<|X|$。我们现在证明这里存在一个与Y有联系、长度为奇数的交变序列。设

$$Y_0=Y-(X\cap Y)$$

$$X_0=X-(X\cap Y)$$

并且G_0为G的诱导子图，它的顶点集为$(X_0\cup Y_0)$。如果G_0是非连通的，那么设G_1，G_2，…，G_k是G_0的连通片，并且$(X_i\cup Y_i)$作为G的顶点集。注意：

$$\bigcup_{i=1}^{k}X_i=X_0$$

和

$$\bigcup_{i=1}^{k}Y_i=Y_0$$

因为$|X|>|Y|$，所以$|X_0|>|Y_0|$。因此对某些i，$|X_i|>|Y_i|$。不失一般性，令$i=1$。

现设T为G的一个生成树。很明显，(X_1, Y_1)是T的顶点集的二划分。因为$|X_1|>|Y_1|$，根据前一个引理，所以在G_1中有一个长度为奇数的极大交变序列σ（对应于Y_1），而且用到了Y_1的所有顶点。很明显，对于G来说，σ也是一个与Y相关联、长度为奇数交变序列。我们现在说明这个σ在G中是极大的序列。

注意：σ的X-顶点和Y-顶点分别属于X_0和Y_0。

考虑一个不在σ中的Y-顶点y_j，这分两种情况。

情况1　$y_j\in Y\cap X$。

在这种情况下，y_j不与X-顶点相邻接。特别是y_j不与σ的任一X-顶点相邻接。

情况2　$y_j\in Y_1$。

在这种情况下，因为σ包含了G_1的所有Y-顶点，所以y_j不在G_1中。因此y_j不与G_1中任一顶点相邻接。特别是y_j不与σ的任一X-顶点相邻接。

这样，在上述两种情况中，y_j都不能用来扩充σ。因此，对于G来说，σ是相应于Y，

长度为奇数的极大交变序列。 □

例如，考虑图9.1中图的独立集$Y=\{a,c\}$，它不是最大的，但可验证$\{d,c,b,a,f\}$是与Y有关、长度为奇数的极大交变序列。

9.2 边覆盖

考虑图$G=(V,E)$，P是E的一个子集，如果G的每个顶点都至少是P中一条边的端点，则称P为G的边覆盖。如果我们认为一条边覆盖了它的端点，那么边覆盖就是覆盖了G的所有顶点的边的子集。

如果G中不存在边覆盖P'，并使$|P'|<|P|$，那么G的边覆盖P称为最小的。G的最小边覆盖中的边数，称为G的边覆盖数，并记为$\beta_1(G)$。

例如，在图9.1所示的图中，集合$\{e_1, e_3, e_5\}$是最小的边覆盖。

回忆得知，顶点覆盖数和边覆盖数分别记为β_0和β_1；类似地独立数和匹配数分别记为α_0与α_1。

下面我们用相同的符号来表示边覆盖以及在覆盖的边上产生的诱导子图。

假设边覆盖P是最小的。那么很容易证实：P既没有长度大于2的回路，也没有这样的路径。这意味着：P的每个连通片都是一个树，而其中所有边都关联在一个公共顶点上。

下面一个定理是推论9.1.1对边的模拟，我们将把α_1和β_1联系起来。这个结论是由文献[9.3]给出的。

定理9.4 对于具有n个顶点而没有孤立顶点的简单图G，

$$\alpha_1+\beta_1=n$$

证明

设M^*为图G的最大匹配，P^*为G的最小边覆盖。令$N(M^*)$表示在M^*中不饱和的顶点集，因此$|N(M^*)|=n-2\alpha_1$。对于$N(M^*)$中的每个顶点，选择这样一条边，使它关联在v上，并且邻接M^*中的一条边。设P_a为这样选择的、$(n-2\alpha_1)$条边的集合。那么很显然，集合$P=M^*\cup P_a$是一个边覆盖，并且

$$|P|=\alpha_1+n-2\alpha_1=n-\alpha_1$$

因此

$$\beta_1=|P^*|\leqslant|P|=n-\alpha_1$$

所以

$$\alpha_1+\beta_1\leqslant n \qquad (9.3)$$

现设P^*具有r个连通片，这样

$$\beta_1=|P^*|=n-r$$

从P的r个连通片的每个连通片中选择一条边，设M是这样选择的r条边的集合。很明显，M是一个匹配，所以

$$\alpha_1=|M^*|\geqslant|M|=r=n-\beta_1$$

因此

$$\alpha_1+\beta_1\geqslant n \qquad (9.4)$$

联立(9.3)式和(9.4)式，可得到

$$\alpha_1 - \beta_1 = n$$
□

在没有孤立顶点的图G中，考虑最大的独立集S^*和最小的边覆盖P^*。因为覆盖S^*的所有顶点需要$|S^*|$条边，所以任一边覆盖必须至少包含$|S^*|$条边。因此，

$$|S^*| \leqslant |P^*| \qquad (9.5)$$

通常，（9.5）式中的等号是不能满足的。然而，当图G是二分图时，所以有$|S^*| = |P^*|$。这将在下一个定理中给予证明，下面定理和定理9.2是类似的。

定理9.5 在一个没有孤立顶点的二分图G中，最大独立集的顶点数等于最小边覆盖的边数，即

$$\alpha_0 = \beta_1$$

证明

设G有n个顶点。那么由推论9.1.1，有$\alpha_0 = n - \beta_0$，再根据定理9.4，有$\beta_1 = n - \alpha_1$。但是按定理9.2，$\alpha_1 = \beta_0$。所以我们得到

$$\alpha_0 = \beta_1$$
□

定理9.2和定理9.5都是Hall定理的等价描述（见习题9.5和9.6）。

9.3 边着色和色指数

一个图的k边着色就是对图中所有边赋于k种不同的颜色。在这种着色中，如果二个相邻接的边不具有相同的颜色，便说这个边着色是完全的。图9.3给出了一个图的3-边着色和完全的4-边着色。如果一个图具有完全的k-边着色，那么这个图就是可k-边着色的。

不失一般性，我们将假定本节中所考虑的图都不具有自环。

一个图G的色指数或边色数$\chi'(G)$为图具有完全k-边着色的最小的k。如果$\chi'(G) = k$，那么图G就是k-边色数的。我们可以验证：图9.3(a)所示图的色指数为4。

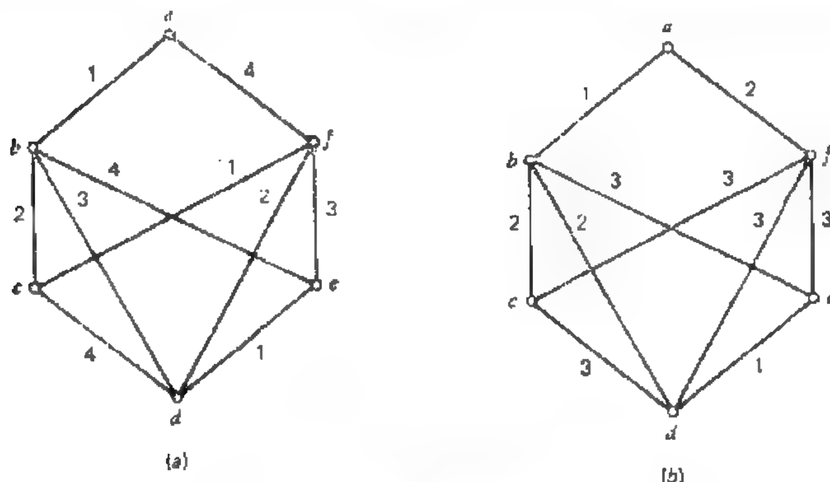


图 9.3 边着色（用数字区别颜色）
(a)完全的4-边着色；(b)3-边着色。

我们可以证明：一个图$G=(V,E)$的k-边着色将引起E的划分（E_1，E_2，…，E_k），其中E_i代表元素在着色中被赋于颜色i的边子集。类似地，若E被划分为k个子集，它相应于G的k-边着色。所以，我们经常用G所引起的划分来表示边着色。

如果一种着色$\mathscr{C}=(E_1, E_2, \cdots, E_k)$是完全的，那么每一个$E_i$都是一个匹配。所以可认为$\chi'(G)$是$G$的边集被划分为匹配的最小数。$\chi'(G)$的这种解释对于证明一些有用的结论是有帮助的。

因为在任何完全的着色中，关联在一个顶点的各边都将得到不同的颜色，所以可推出

$$\chi'(G)\geqslant\Delta \tag{9.6}$$

这里Δ是G中顶点的最大度。

通常$\chi(G)\neq\Delta$。然而在二分图中，$\chi'=\Delta$。

定理9.6 对于一个二分图，

$$\chi'=\Delta$$

证明

根据推论8.22.1，二分图的边集能被划分为Δ个匹配。因此

$$\chi'(G)\leqslant\Delta$$

联立此式和(9.6)式，定理得证。 □

在一般情况下，虽然用Δ种颜色给G的边完全地着色是不充分的，但是文献[9.4]曾指出：对任一简单图，$\Delta+1$种颜色是充分的。为了证明Vizing定理，我们须要证实一些基本结论。

在一种着色中，如果关联在v上的边至少有一条被赋于颜色i，那么我们就说，颜色i代表顶点v。$c(v)$用来表示顶点v的不同颜色数。例如，在图9.3(b)的着色中，

$$c(f)=2$$

我们现在考虑图存在2-边着色的问题，即图中每个顶点v都有$c(v)=2$。

引理9.3 设G不是一个奇数长度回路的连通图。那么G就有一个2-边着色，其中两种颜色都出现在每个度数至少为2的顶点上。

证明

情形1 G是欧拉图。

如果G是一个回路，那么按定理的前提，它必须是偶数长度。在这种情况下，很容易验证：G具有所要求的2-边着色性质。

如果G不是一个回路，那么必须有一个度数至少为4的顶点v_0。令

$$v_0, e_1, v_1, e_2, v_2, e_3, \cdots, e_m, v_0$$

为欧拉轨迹，并且令

$$E_1=\{e_i \mid i\text{ 为奇数}\},\ E_2=\{e_i \mid i\text{ 为偶数}\} \tag{9.7}$$

那么(E_1, E_2)是一个2-边着色，它的每个顶点都有两种颜色出现，这是因为在所考虑的欧拉轨迹中，每个顶点(包括v_0)都视为内部顶点。

情形2 G不是欧拉图。

在这种情况下，G必须具有偶数个奇度数的顶点。现在我们来构造一个欧拉图G'，加一个新顶点v_0，并把它与图G中每个奇数度的顶点相连接。设$v_0, e_1, v_1, \cdots, e_m, v_0$是$G'$中的一条欧拉轨迹，如果$E_1$和$E_2$的定义如(9.7)式，那么$(E_1, E_2)$就是一个$G'$的2-边着色，其中两种颜色都出现在图$G$的每个顶点上。可以验证：$(E_1\cap E, E_2\cap E)$是

一个G的2-边着色，其中两种颜色都出现在每个度数至少为 2 的顶点上。 □

对于图$G=(V, E)$的k-边着色$\mathscr{E}$，如果不存在另外一种k-边着色$\mathscr{E}'$，以使得

$$\sum_{v\in V}c(v)<\sum_{v\in V}c'(v)$$

这里$c(v)$和$c'(v)$分别表示在$\mathscr{E}$和$\mathscr{E}'$中出现在顶点上的不同颜色数目。那么图$G=(V, E)$的k-边着色$\mathscr{E}$就是最优的。

很显然，在任一种着色$\mathscr{E}$中，对于每一个顶点v，$c(v)\leqslant d(v)$。进而，对于每个v，$c(v)=d(v)$，当且仅当$\mathscr{E}$是一种完全的着色。

下面介绍一个关于最优着色属性的有用结论。

引理9.4 让$\mathscr{E}=(E_1, E_2, \cdots, E_k)$是图$G=(V, E)$的一种最优$k$-边着色。假设在$G$中存在一个顶点$u$，并且有两种颜色$i$和$j$，其$i$不出现在$u$上，而$j$出现在$u$上两次。设$G'$是$G$在边集$E_i\cup E_j$上的诱导子图，那么包含顶点$u$的$G'$的片$H$就是奇数长度回路。

证明

如H不是奇数长度回路，那么按引理9.3，则存在一个H的2-边着色，它的两种颜色i和j都出现在H中度数至少为 2 的每个顶点上。

如果我们使用这种2-边着色重新给H的边着色，并使G中另外一些边的颜色和它们在$\mathscr{E}$中颜色保持不变，那么我们将得到一种新的k-边着色$\mathscr{E}'$，在其中两种颜色i和j都出现在顶点u上。所以

$$c'(u)=c(u)+1$$

这里$c'(u)$是在$\mathscr{E}'$中表示顶点u的不同颜色数。然而，对于每一个顶点$v\neq u$，

$$c'(v)\geqslant c(v)$$

所以

$$\sum_{v\in V}c'(v)>\sum_{v\in V}(v)$$

这与$\mathscr{E}$是一种最优边着色相矛盾。因此，可以说明：H是奇数长度的回路。 □

如果

$$\sum_{v\in V}c'(v)>\sum_{v\in V}c(v)$$

那么一种k-边着色$\mathscr{E}'$就是k-边着色$\mathscr{E}$的一种改进。

我们现在证明定理[9.4]。我们的证明是由文献[9.5]给出的。我们采用由文献[9.6]给出的论述。

定理9.7（VIZING） 如果$G=(V, E)$是个简单图，那么或者$\chi'(G)=\Delta$，或者$\chi'(G)=\Delta+1$。

证明

因为$\chi'(G)\geqslant\Delta$，所以我们只需证明$\chi'(G)\leqslant\Delta+1$。

设$\mathscr{E}=\{E_1, E_2, \cdots, E_{\Delta+1}\}$是$G$的一种最优$(\Delta+1)$-边着色。假定$\chi'(G)>\Delta+1$，即$\mathscr{E}$不是完全的$(\Delta+1)$-边着色。那么在$G$中就存在一个顶点$u$，使得$c(u)<d(u)$。这意味着，这里存在着颜色$i_0$和$i_1$，使其$i_0$不出现在$u$上，而$i_1$出现在$u$上两次。设边$(u, v_1)$具有颜色$i_1$（图9.4(a)）。

因为在G中最大的度为Δ，它说明：在着色$\mathscr{C}$中，不是所有的$\Delta+1$种颜色都出现在v_1上。

设颜色i_2不出现在顶点v_1上。那么i_2就必须出现在顶点u上。否则，我们就能用颜色i_2对边（u, v_1）重新着色，并且得到了G的一种新的着色，这种新着色就是对$\mathscr{C}$的一种改进，令边（u, v_2）具有颜色i_2。再重复一次，不是所有的颜色都出现在v_2上。

假设颜色i_3不出现在v_2上，那么它必出现在u上，否则，我们就能用颜色i_2对边（u, v_1）重新着色，用颜色i_3对边（u, v_2）重新着色，因此得到了$\mathscr{C}$的一种改进。令边（u, v_3）具有颜色i_3。

重复上面的论述，我们能构造一个顶点序列v_1, v_2, …和一个颜色序列i_1, i_2, …，使得

1.（u, v_j）具有颜色i_j;

2. i_{j+1}不出现在v_j上。

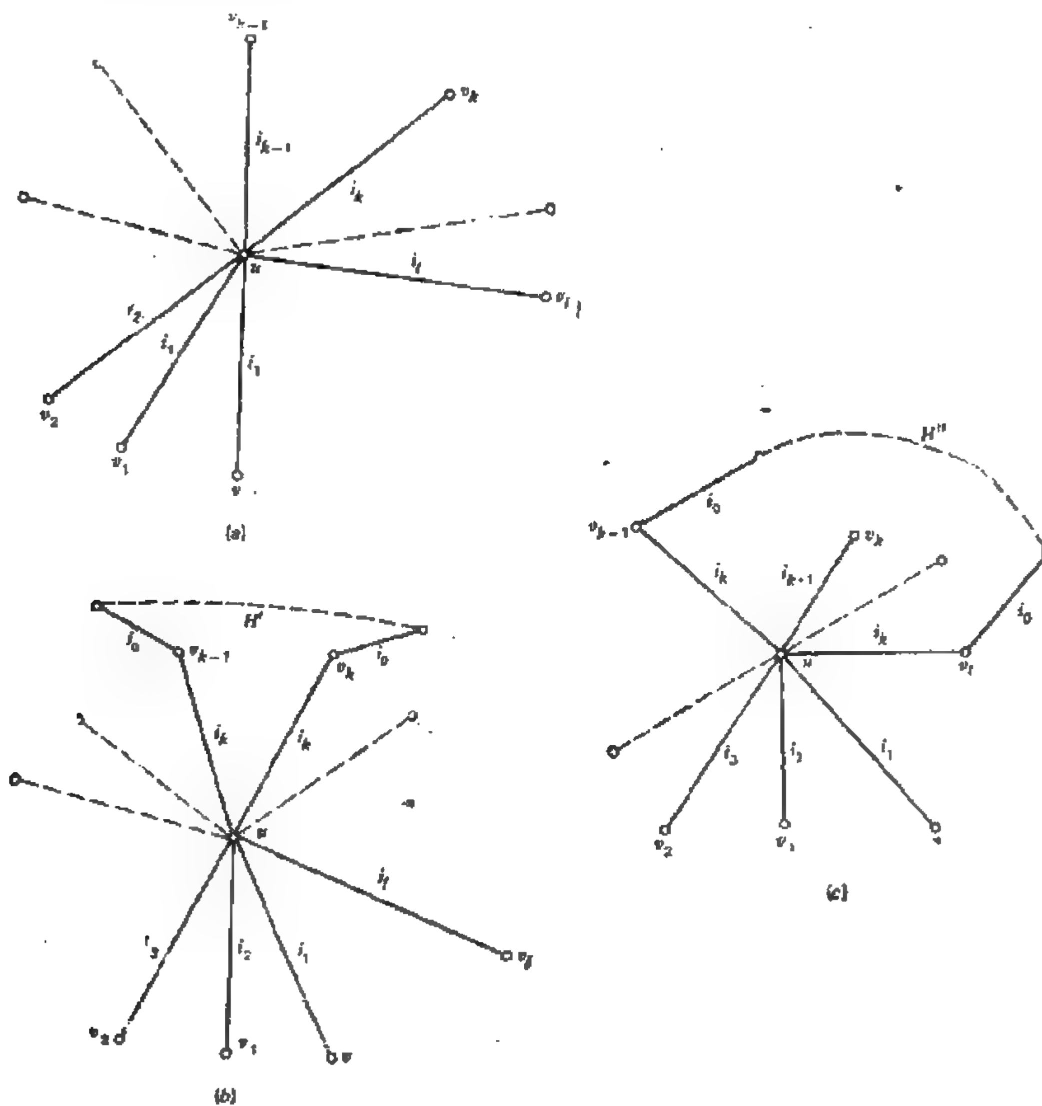


图 9.4

现在我们很容易地看出：这里存在一个最小的整数 l，对一些 $k<l$，使得

$$i_{l+1}=i_k \qquad （图9.4(a)）$$

为了说明矛盾性，我们下面用两种不同的方法来重新着色关联在u上的一些边，并得到两种新的、最优$(\Delta+1)$-边着色$\mathscr{E}'$和$\mathscr{E}''$。

$\mathscr{E}'=\{E'_1, E'_2, \cdots, E'_{\Delta+1}\}$是用颜色$i_{j+1}$重新对（$u$, v_j）进行着色而得到的，$1\leqslant j\leqslant k-1$。$G$中余下来的每条边在$\mathscr{E}'$中具有和在$\mathscr{E}$中相同的颜色（图9.4(b)）。

$\mathscr{E}''=\{E''_1, E''_2, \cdots, E''_{\Delta+1}\}$是用颜色$i_{j+1}$重新对（$v$, v_j）着色而得到的，$1\leqslant j\leqslant l-1$，并使（$u$, v_l）具有颜色i_k。G中每个余下来的边在$\mathscr{E}''$中具有和在$\mathscr{E}$中相同的颜色（图9.4（c））。

设G'是G的一个子图，其边集为$(E'_{i_0}\cup E'_{i_k})$，并且用H'来表示包含u的G的一个片。同样，G''是G的子图，其边集为（$E''_{i_0}\cup E''_{i_k}$），并且用H''来表示含u的G的一个片。

在$\mathscr{E}'$和$\mathscr{E}''$中，颜色i_0不出现在u上，而颜色i_k出现在这个顶点上两次。所以根据引理9.4可知，H'和H''都是回路。

现在（u, v_k）是H'中仅有的边，而不在H''中。所以u和v_k在G''中也是连通的。这样它们位于G''的相同片上，即H''片上。在H''中，v_k的度数为1，否则，它在H'中的度数将大于2。这与H''也是回路相矛盾。因此，$\mathscr{E}$是一个完全的$(\Delta+1)$-边着色。 □

比上述定理更一般的结论，参见习题9.8。

9.4 顶点着色和色数

一个图的k-顶点着色就是对图的各顶点赋于k种不同的颜色。如果在一种着色中，相邻接的顶点不具有相同的颜色，那么这个顶点着色就是完全的。在图9.5中给出了一种完全的3-顶点着色。如果一个图具有一种完全k-顶点着色，那么这个图就是可k-顶点着色的。

不失一般性，我们假设本节所考虑的图都是简单图。

一个图G的色数$\chi(G)$是当G为可k-顶点着色时的最小数k。如果$\chi(G)=k$，则G是k-着色。例如，图9.5所示图的色数为3。

从此以后，我们把"完全的k-顶点着色"简化为"k-着色"。类似地，将"可k-顶点着色"简化为"可k-着色"。

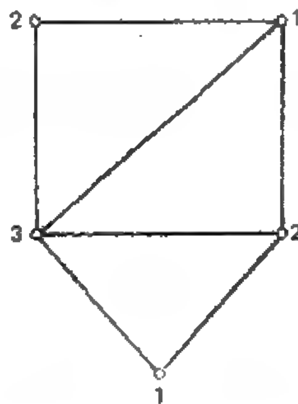


图 9.5 完全的3-顶点着色

注意，一个图$G=(V, E)$的一种k-着色可引起对V的一个划分$(V_1, V_2, \cdots, V_k)$，这里每个V_i都是一个顶点子集，这个子集中的元素被赋于颜色i，因此它是独立集。类似地，若V被划分为k个独立集，相应于G的一种k着色。

在前一节（定理9.7）已经证明：对简单图的边进行完全地着色最多需要$\Delta+1$种颜色。将证明对于顶点着色也有类似的结论。

定理9.8 如果图G是简单的，那么它就是可$(\Delta+1)$

一着色的。

证明

若给定$\Delta+1$种不同的颜色，我们按如下方法能得到一种G的$(\Delta+1)$-着色。

对任一顶点v_0赋于它$\Delta+1$种颜色之一。然后选择一个未着色的顶点（比如说v_1），赋于v_1一种未赋给与它邻接顶点的颜色。这总是可能的，因为$d(v_1)\leqslant\Delta$，所以至多有Δ种颜色已赋给与v_1相邻接的顶点。重复这个进程，直到所有的顶点都被着色为止。这样得到的着色显然是一种完全的$(\Delta+1)$-着色。 □

很显然，对于完全图和长度为奇数的回路，$\chi=\Delta+1$。一个非常有趣的结论是，对于所有其它的图，$\chi\leqslant\Delta$。这个结论是由文献[9.7]给出的，以后再证。我们这里的证明是文献[9.8]给出的。另外一种不同的证明可见文献[9.9]。

定理9.9（BROOKS） 设G是一个连通的简单图。如果G既不是完全图也不是长度为奇数的回路的话，那么$\chi(G)\leqslant\Delta$。

证明

很显然，对于$\Delta=0$、1、2，定理成立。

为了证明定理对$\Delta\geqslant3$成立，采用反证法，即设存在一类图，它不是完全图，并且$\Delta\geqslant3$，$\chi=\Delta+1$。然后选择一个具有最小顶点数的图$G=(V, E)$。

设$v_0\in V$，G'为从G中移去v_0而得到的图。这说明，从G的选择得到的G'是可Δ-着色的，这意味着：$d(v_0)=\Delta$，否则，用于着色G'的Δ种颜色之一就能够给v_0进行着色，但这与$\chi(G)=\Delta+1$相矛盾。另外一些重要的含义如下：

性质1 在G'的任何Δ-着色中，与v_0相邻接的顶点被着以不同的颜色。

设u_1，u_2，…，u_Δ是与v_0相邻接的顶点。设u_1，u_2，…，u_Δ在G'的着色中分别赋于颜色1，2，…，Δ。设$G(i, j)$表示G'的诱导子图，它的顶点被赋于颜色i和j。然后，

性质2 顶点u_i和u_j在$G(i, j)$的相同的连通片中。

否则，在包含u_i的片中互换颜色i和j，我们能得到一种新的G'的Δ-着色，其中u_i和u_j赋于相同的颜色，但这与性质1相矛盾。

设C_{ij}是包含u_i和u_j的$G(i, j)$的片。然后，

性质3 C_{ij}是从u_i到u_j的一条路径。

假设C_{ij}中u_i的度大于1，那么u_i至少和二个颜色为j的顶点相邻接。因为在G'中，$d(u_i)\leqslant\Delta-1$，所以我们能用颜色$k\neq i$、j给u_i重新着色，以致于产生一种新的着色，使u_i和u_k具有相同颜色，但这与性质1相矛盾。

类似地，C_{ij}中的u_j的度也为1。

C_{ij}中所有其它顶点度都为2。否则，当我们沿路径从u_i到u_j移动时，设u是第一个度数（在C_{ij}中）大于2的顶点。如果u用颜色i来着色，那么它至少与三个颜色为j的顶点相邻接。因为$d(u)\leqslant\Delta$，所以我们能用颜色$k\neq i$、j给u重新着色，以致于在新的着色中，u_i和u_j在不同的片中，但这与性质2相矛盾。

因此，C_{ij}是从u_i到u_j的一条路径。

性质4　C_{ij}和C_{ik}除u_i外设有公共顶点。

设$u\neq u_i$是C_{ij}和C_{ik}中的公共顶点。然后u用颜色i进行着色，并且至少与两个颜色为j的顶点和两个颜色为k的顶点相邻接。因为，$d(u)\leqslant\Delta$，所以存在一种颜色$l\neq i$、j、k，用它给u重新着色。但这将使u_i和u_j不连通，而且与性质2相矛盾。

现在我们推导一个与性质4相矛盾的论述。

因为G不是一个$(\Delta+1)$-顶点的完全图，则存在两个不相邻接的顶点，比如u_1和u_2。那么路径C_{12}包含了一个与u_1相邻接的顶点$u\neq u_2$。假设在路径C_{13}中（因$\Delta\geqslant3$，故C_{13}存在），互换颜色1和3，使得在新的着色C'中，u_1得到颜色3，而u_3得到颜色1，但是在新的片C'_{12}和C'_{23}中包含公共顶点$u\neq u_2$，这与性质4相矛盾。

这样就完全地证明了定理9.9。 □

9.5 色多项式

本节将讨论计算图的不同的完全λ-着色数目问题。

如果一个图是可λ-着色的，那么就能够用λ种颜色、不止一种方法给它着色，如果在两种着色中，一个图至少有一个顶点被赋于不同的颜色，那么可认为这个图的两种着色是不同的。

对于每个整数λ，色多项式$P(G,\ \lambda)$表示了图G可能有的、不同λ-着色的数目。

例如，考虑图9.6。若给定λ种颜色，那么我们能选择其中一种给顶点a着色。然后，可用剩下的$\Delta-1$种颜色之一给顶点b着色。对于顶点b的每种着色，都有对顶点c进行着色的λ-1种不同方法。因此图9.6所示的图能用$\lambda(\lambda-1)^2$种不同的方法给予着色。换言之，

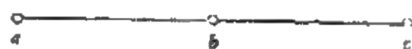


图 9.6

这个图的色多项式为$\lambda(\lambda-1)^2$。事实上，我们可重复这样的结论来证明：具有n个顶点的路的色多项式为$\lambda(\lambda-1)^{n-1}$。

做为另外一个例子，考虑具有n个顶点$v_1,\ v_2,\ \cdots,\ v_n$的完全图K_n。若给定λ种颜色，那么顶点v_1能用λ种颜色之一着色，顶点v_2能用剩下的$\lambda-1$种颜色之一着色，顶点v_3能用剩下的$\lambda-2$种颜色之一着色，如此下去。因此，

$$P(K_n,\ \lambda)=\lambda(\lambda-1)\cdots(\lambda-n+1)$$

下面我们介绍确定一个图G色多项式的公式。

定理9.10　设u和v在简单图G中是二个不相邻接的顶点，并设$e=(u,\ v)$，如果$G\cdot e$表示在G中使u和v短路并用单边代替平行边后的简单图，而且$G+e$表示在图G中加上边e而得到的图，那么

$$P(G,\ \lambda)=P(G+e,\ \lambda)+P(G\cdot e,\ \lambda)$$

证明

在每个G的λ-着色中，G中顶点u和v被赋予对应于$G+e$的一种λ-着色的不同颜色，反之亦然。类似地，在每个G的λ-着色中，G中u和v被赋予对应于$G\cdot e$的一种λ-着色的相同颜色，反之亦然。所以

$$P(G,\ \lambda)=P(G+e,\ \lambda)+P(G\cdot e,\ \lambda)$$ □

以上结论可表述为如下的不同形式。

推论9.10.1 如$e=(u,\ v)$是简单图的一条边，那么

$$P(G,\ \lambda)=P(G-e,\ \lambda)-P(G\cdot e,\ \lambda)$$

这里$G-e$是从G中移去e而得到的图，而$G\cdot e$和定理9.10中所定义的一样。 □

我们如果在一个图G上反复地应用定理9.10给出的公式，那么这个过程将终止到完全图，譬如H_1，H_2，…，H_k。这样有：

$$P(G,\ \lambda)=P(H_1,\ \lambda)+P(H_2,\ \lambda)+\cdots+P(H_k,\ \lambda)$$

另一方面，如果我们用推论9.10.1给出的公式，那么这个过程将终止到空图（即没有边的图），并使得色多项式由空图的色多项式的线性组合而成。

图9.7说明了上述两个过程。

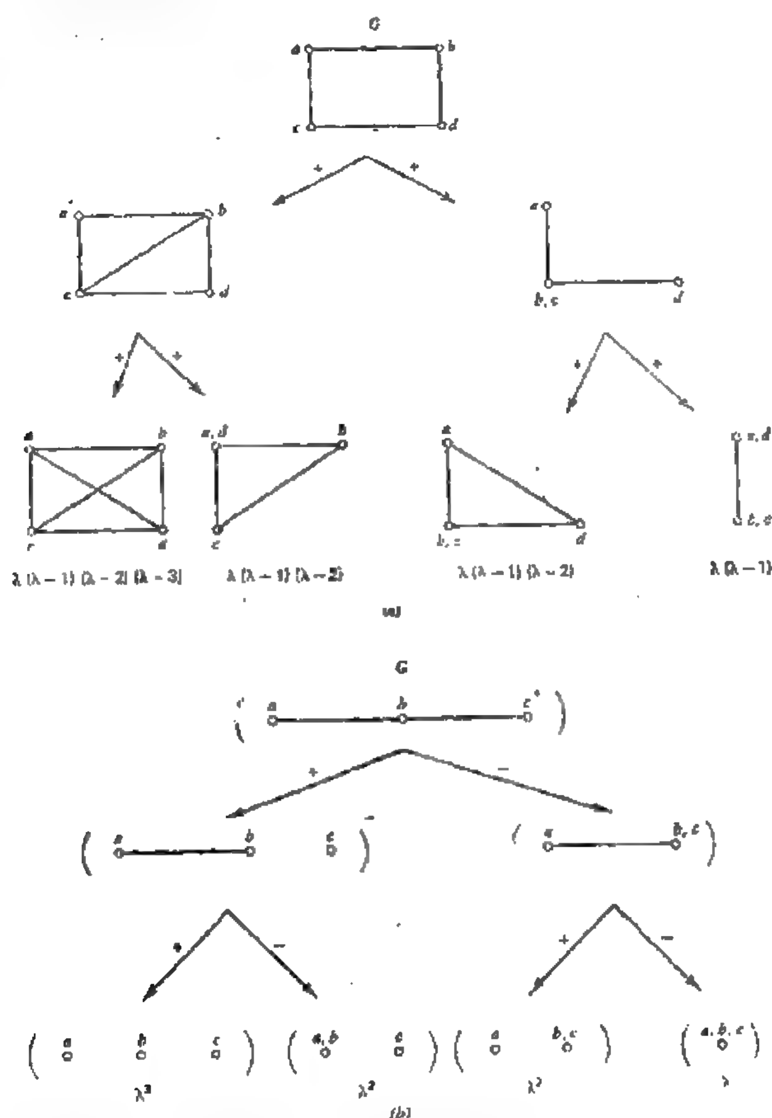


图 9.7 图G色多项式的计算

(a)利用定理9.10，$P(G,\ \lambda)=\lambda(\lambda-1)(\lambda-2)(\lambda-3)+2\lambda(\lambda-1)(\lambda-2)+\lambda(\lambda-1)$；

(b)利用推论9.10.1，$P(G,\ \lambda)=\lambda^3-2\lambda^2+\lambda$。

定理9.11 具有n个顶点的图G，其色多项式$P(G, \lambda)$具有n阶，其中开始项为λ^n，常数项为零。因此，它的系数都是整数，并交替变换符号。

证明

这个证明是用对边数m进行归纳得到的。很显然，$m=0$时，定理成立，因为n个顶点的空图色多项式为λ^n。

假设对于边数小于m的所有图，定理成立。现考虑m条边、n个顶点的图。设e为图的一条边。那么$G-e$是具有$m-1$条边、n个顶点的图，而$G\cdot e$是具有$(n-1)$个顶点图，其边为$m-1$或更少些。

根据归纳前提，这可得出非负的整数系数a_1，a_2，…，a_{n-1}和b_1，b_2，…，b_{n-2}，使得

$$P(G-e, \lambda)=\lambda^n-a_{n-1}\lambda^{n-1}+a_{n-2}\lambda^{n-2}-\cdots+(-1)^{n-1}a_1\lambda$$

和

$$P(G\cdot e, \lambda)=\lambda^{n-1}-b_{n-2}\lambda^{n-2}+b_{n-3}\lambda^{n-3}-\cdots+(-1)^{n-2}b_1\lambda$$

根据推论9.10.1，

$$\begin{aligned}P(G, \lambda)&=P(G-e, \lambda)-P(G\cdot e, \lambda)\\&=\lambda^n-(a_{n-1}+1)\lambda^{n-1}+\sum_{i=2}^{n-1}(-1)^i(a_{n-i}+b_{n-i})\lambda^{n-i}\end{aligned}$$

因此，G也满足此定理。

9.6 四色问题

绘制地图时，为了区别不同的区域，人们对区域着色发生了兴趣，以使两个相邻的区域具有不同颜色。这就是一个给平面图区域进行完全着色的问题，它与其对偶图的顶点完全着色是同一问题。人们很容易举例来说明：通常的三色不足以给一个平面图进行完全地着色。在下面被称为五色定理中，我们将证明五种颜色是充分的。

定理9.12 每一个平面图是可5-着色的。

证明

假设对于所有阶数小于n的平面图，定理为真，那么再证明对于阶数为n的平面图，定理亦为真。

根据推论7.3.5，在G中存在一个顶点v_0，其度$\leqslant 5$。设G'是在$V-\{v_0\}$上的G的诱导子图。

用五种颜色α_1，α_2，α_3，α_4，α_5给G'着色（按归纳假设这是可能的），很显然，在G的完全5-着色中，除非$d(v_0)=5$，否则都能将这些颜色之一赋于v_0，并且对于邻接v_0的所有五个顶点v_1，v_2，…，v_5都赋于不同的颜色。

假设v_i赋于颜色α_i，$1\leqslant i\leqslant 5$。进而，设v_1，v_2，…，v_5按顺时排列，如图9.8所示。

设$G(\alpha_i, \alpha_j)$是G的一个子图，其顶点被赋于颜色α_i或α_j。

在$G(\alpha_1, \alpha_3)$中，包含v_1的片必然包含v_3；否则，颜色α_1和α_3在这个片中可交换，并且v_0可用α_1着色。

类似地，在$G(\alpha_1, \alpha_4)$中，包含v_1的片面也必然包含v_4（见图9.8）。

那么在$G(\alpha_2, \alpha_5)$中，v_2和v_5将是不连通的。如果在包含v_2的$G(\alpha_2, \alpha_5)$的片中互换颜色α_2和α_5，那么v_0将可用α_2着色。因此，G是可5-着色的。 □

上述结论是由文献[9.10]给出的。

现在提出这样一个问题，是否五色定理是可能中最好的。人们曾猜想过：每个平面图是可4-着色的。这就是众所周知的四色猜想。这个问题100多年来仍未解决，成为一个广泛研究的课题。最近，文献[9.11]证明了这个猜想为真，其表述为如下定理。

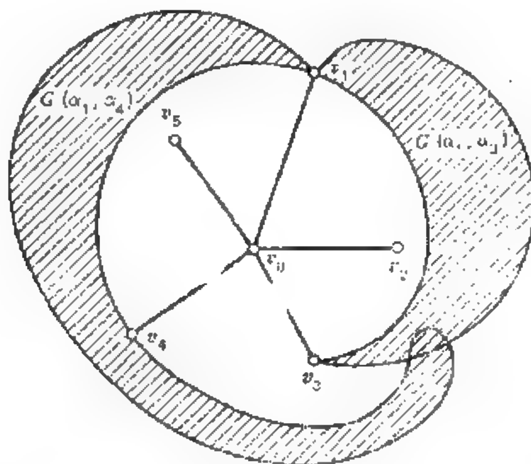


图 9.8

定理9.13（四色定理） 每一个平面图是可4-着色的。 □

在四色问题上，有浩繁大量的文献。在文献[9.12]和[9.13]以及[9.14]中都作了历史细节的广泛综述。还可参阅文献[9.15]。

9.7 进一步阅读

文献[9.2]、[9.6]、[9.16]为一般性的参考文献，但都涉及到了本章的题目。

在图论中，一个由文献[9.17]给出的经典结论，确定了具有n个顶点和独立数$\leqslant k$的图的最小边数（习题9.2），参见文献[9.18]。这个结论是图论一个分支的起点，称为极端图论。文献[9.19]是专门研究极端图的问题，极端结论形成了设计具有特定抗扰性和可靠性通讯网络的基础，可参见文献[9.20]、[9.21]、[9.22]和[9.23]。

关于独立数在信息论中的应用，参见文献[9.2]（第十六章）、[9.24]（第九章）和[9.25]。

文献[9.26]给出了一个关于色多项式的全面综述，这也可参阅[9.27]和[9.28]、[9.29]。文献[9.24]（pp.245～246）讨论了着色在把一个图分解为平面子图中的应用。这个问题是在印刷电路设计中提出来的。参看文献[9.30]，它是关于着色在时间表问题中的应用，以及文献[9.31]关于着色的算法。

在图G中，设有顶点集合D，如果不在D中的每个顶点都与D中一个顶点相邻接，则D称为G的控制集。为了复习关于控制集的某些结论，可参阅文献[9.32]。控制集的极端结论在文献[9.33]中有论述。

9.8 习　　题

9.1 证明或证伪：一个图的每个顶点覆盖都包含一个最小顶点覆盖。

9.2 若给定任意两个整数n和k，且$n\geqslant k>0$。设$q=\lfloor n/k\rfloor$和r是一个整数，有

$$n=kq+r,\quad 0\leqslant r<k$$

设$G_{n,k}$是由k个不相交的完全图组成的简单图，其中r个图有$q+1$个顶点，$k-r$个图有q个顶点。然后证明：每个具有n个顶点和$\alpha_0(G)\leqslant k$、且具有最小可能边数的图G同构于$G_{n,k}$(文献[9.17])。

这个结论的简单推论如下：

如果G是具有n个顶点、m条边的简单图，并且$\alpha_0(G)=k$，那么

$$m\geqslant q\left(n-\frac{k}{2}-\frac{k_q}{2}\right)$$

9.3 如果图G是具有n个顶点、m条边的简单图，那么试证明：

$$\alpha_0(G)\geqslant\frac{n^2}{2m+n}$$

9.4 试证明：具有n个顶点而没有三角形的简单图G，其边的最大数可为$\lfloor n^2/4\rfloor$。根据这个性质构造一个图。

9.5 根据定理9.2，试推导出Hall 定理，即定理8.13。

9.6 根据定理9.5，试推导出Hall 定理，即定理8.13。

9.7 证明：

$$\chi'(K_n)=\begin{cases}n-1, & \text{如果 } n \text{ 为偶数}\\ n, & \text{如果 } n \text{ 为奇数}\end{cases}$$

9.8 证明：如G是无自环的图，那么

$$\chi'(G)\leqslant\Delta+k$$

这里k是G的任意两个顶点间最大平行边数（文献[9.4]）。

9.9 设G是无自环的图。如果$\Delta(G)=3$，试证明：$\chi'(G)=3$或4。

9.10 证明：对于奇数顶点的非空正则简单图G，有

$$\chi'(G)=\Delta+1$$

9.11 证明：对于任意给定边方向的简单图G，总存在一条长度为$\chi(G)-1$的有向路径。

9.12 证明：如果图G中任何两个奇长度回路有一个公共顶点，那么

$$\chi(G)\leqslant 5$$

9.13 利用推论7.3.5，证明每个平面图是可6-着色的

9.14 证明：如果一个简单图G具有度序列$(d_1,\ d_2,\ \cdots,\ d_n)$并且$d_1\geqslant d_2\geqslant\cdots\geqslant d_n$，那么

$$\chi(G)\leqslant\max_i\ \min\{d_i+1,\ i\}$$

（见文献[9.34]）

9.15 证明：对于具有n个顶点的简单图G，

(a) $\chi(G)\alpha_0(G)\geqslant n$;

(b) $\chi(G)+\alpha_0(G)\leqslant n+1$;

(c) $\chi(G)+\chi(\overline{G})\leqslant n+1$。

9.16 设G是一个简单的具有n个顶点、且度为k的正则图。试证明：

$$\chi(G) \geqslant \frac{n}{n-K}$$

9.17 设G是一个平面图，并且它的每个域恰被三条边所包围。

试证明：G是可3-着色的，当且仅当G是欧拉图。

9.18 证明：如果一个图具有7个顶点、而顶点度都是4，那么它就是非平面图。（文献[9.24]）

9.19 （a）试证明：一个图是可2-着色的，当且仅当它的所有回路都具有偶数长度。

（b）试证明：一个平面图G的域能被二种颜色完全地着色，当且仅当图G是欧拉图。

9.20 求出图9.9所示图的色多项式。

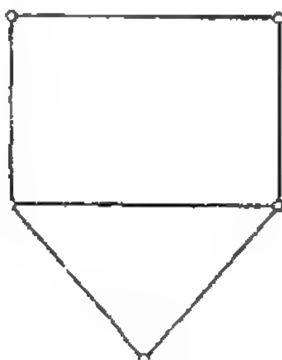


图 9.9

9.21 （a）试证明：具有n个顶点的图G是一个树，当且仅当

$$P(G, \lambda)=\lambda(\lambda-1)^{n-1}$$

（b）试证明：如果具有n个顶点的图G是连通的，那么

$$P(G, \lambda)\leqslant\lambda(\lambda-1)^{n-1}$$

9.22 试证明：如果C是长度为n的回路，那么

$$P(G, \lambda)=(\lambda-1)^n+(-1)^n(\lambda-1)$$

9.23 试证明：如果G是具有$n+1$个顶点的轮形图，那么

$$P(G, \lambda)=\lambda(\lambda-2)^n+(-1)^n\lambda(\lambda-2)$$

9.24 试证明：如果一个简单图G具有n个顶点、m条边和k个连通片，那么

（a）在$P(G, \lambda)$中，λ^{n-1}的系数为$-m$。

（b）在具有非零系数的$P(G, \lambda)$中，λ的最小指数是k。

9.25 一个图$G=(V,E)$的团是个集合$S\subseteq V$，而且在S上，G的诱导子图是个完全图。设$P=\{P_1,P_2,\cdots,P_k\}$是V划分为团的一个。设

$$\theta(G)=\min_{P}\{|P|\}$$

则$\theta(G)$是V划分为团的可能最小数。

试证明：对于一个简单图G，$\alpha_0(G)\leqslant\theta(G)$。进而，如果$S$是一个独立集，$P$是$V$的一个划分为团，且$|S|=|P|$，那么证明：$S$是最大独立集，$P$是$V$划分为团的最小数。

9.9 参考文献

9.1 D. König, "Graphs and Matrices," (in Hungarian), *Mat. Fiz Lapok*, Vol. 38, 116—119(1931).

9.2 C. Berge, *Graphs and Hypergraphs*, North Holland, Amsterdam, 1973.

9.3 T. Gallai, "Über extreme Punkt und Kantenmengen," *Ann. Univ. Sci. Budapest, Eötvös Sect. Math.*, Vol. 2, 133—138(1959).

9.4 V. G. Vizing, "On an Estimate of the Chromatic Class of a p-Graph," (in Russian), *Diskret. Analiz.*, Vol. 3, 25—30(1964).

9.5 J. C. Fournier, "Colorations des arêtes d'un graphe," *Cahiers du CERO*, Vol. 15, 311—314(1973).

9.6 J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, Macmillan, London, 1976.

9.7 R. L. Brooks, "On Colouring the Nodes of a Network," *Proc. Cambridge Phil. Soc.*, Vol. 37, 194—197(1941).

9.8 L. S. Melnikov and V. G. Vizing, "New Proof of Brooks' Theorem," *J. Combinatorial Theory*, Vol.7, 289—290 (1969).

9.9 L. Lovász, "Three Short Proofs in Graph Theory," *J. Combinatorial Theory B*, Vol. 19, 111—113(1975).

9.10 P. J. Heawood, "Map Colour Theorems," *Quart. J. Math.*, Vol. 24, 332—338 (1890).

9.11 K. I. Appel and W. Haken, "Every Planr Map is Four-Colorable," *Bull. Am. Math. Soc.*, Vol. 82, 711—712(1976).

9.12 O.Ore, *The Four—Color Problem*, Academic Press, New York, 1967.

9.13 T. L. Saaty, "Thirteen Colorful Variations on Guthrie's Four-Color Conjecture," *Am. Math. Monthly*, Vol. 79, 2—43(1972).

9.14 T. L. Saaty and P. C. Kainen, *The Four-Color Problem: Assaults and Conquests*, McGraw-Hill, New York, 1977.

9.15 H. Whitney and W. T. Tutte, "Kempe Chains and the Four Colour Problem," in *Studies in Graph Theory*, Part II, MAA Press, 1975, pp. 378—413.

9.16 F. Harary, *Graph Theory*, Addison-Wesley, Reading, Mass., 1969.

9.17 P. Turán, "An Extremal Problem in Graph Theory," (in Hungarian), *Mat. Fiz. Lapok*, Vol. 48, 436—452(1941).

9.18 P. Erdös, "On the Graph Theorem of Turán," (in Hungarian), *Mat. Lapok*, Vol. 21, 249—251(1970).

9.19 B. Bollobás, *Extremal Graph Theory*, Academic Press, New York, 1978.

9.20 H. Frank and I. T. Frisch, *Communication, Transmission, and Transportation Networks*, Addison-Wesley, Reading, Mass., 1971.

9.21 F. T. Boesch (Ed.), *Large-Scale Networks: Theory and Design*, IEEE Press, New York, 1976.

9.22 P.Karivaratharajan and K.Thulasiraman,"K-Sets of a Graph and Vulnerability of Communication Nets," *Matrix and Tensor Quart.*, Vol. 25,63—66, (1974), 77—86(1975).

9.23 P Karivaratharajan and K. Thulasiraman "An Extremal Problem in Graph

Theory and Its Applications," *Proc. IEEE Intl. Symp. on Circuits and Systems*, Tokyo, Japan, 1970.

9.24 C. L. Liu, *Introduction to Combinatorial Mathematics*, McGraw-Hill, New York, 1968

9.25 L. Lovász, "On the Shannon Capacity of a Graph," *IEEE Trans. Inform. Theory*, Vol. IT-25, 1—7(1979).

9.26 R. C. Read, "An Introduction to Chromatic Polynomials," *J. Combinatorial Theory*, Vol. 4, 52—71(1968).

9.27 G. D. Birkhoff, "A Determinant Formula for the Number of Ways of Coloring a Map," *Ann. Math.*, Vol. 14, 42—46(1912).

9.28 W. T. Tutte, "On Chromatic Polynomials and the Golden Ratio," *J. Combinatorial Theory*, Vol. 9, 289—296(1970).

9.29 W. T. Tutte, "Chromials," in *Studies in Graph Theory*, Part II, MAA Press, 1975, pp. 361—377.

9.30 D. C. Wood, "A Technique for Colouring a Graph Applicable to Large-Scale Timetabling Problems," *Computer J.*, Vol. 12, 317(1969).

9.31 N. Christofides, *Graph Theory: An Algorithmic Approach*, Academic Press, New York, 1975.

9.32 E. J. Cockayne and S. T. Hedetniemi, "Towards a Theory of Domination in Graphs," *Networks*, Vol. 7, 247—257(1977).

9.33 E. J. Cockayne and S. T. Hedetniemi, "Optimal Domination in Graphs," *IEEE Trans. Circuits and Sys.*, Vol. CAS-22, 41—44(1975).

9.34 D. J. A. Welsh and M. B. Powell, "An Upper Bound for the Chromatic Number of a Graph and Its Application to Timetabling Problems," *Computer J.*, Vol. 10, 85—87(1967).

第十章　拟　阵

考虑任一域上的矢量有限集S。众所周知，S的每个矢量子集或者线性相关，或者线性独立。进而，矢量的独立集聚合有几个有趣的性质。例如，

1.独立集的任何子集都是独立的。

2.如果I_p和I_{p+1}是任意两个独立集，而且$|I_{p+1}|=|I_p|+1$，那么I_p和I_{p+1}的某些元素一起构成了具有$|I_{p+1}|$个元素的独立集。

有些代数系统具有上述性质。例如，不含回路的图的边子集的聚合就具有上述性质。正当在研究这类系统的性质时，Whitney在文献[10.1]中引入了拟阵的概念。

本章将对拟阵理论作初步介绍，讨论拟阵的一些基本性质。我们特别感兴趣的是图的回路和割集之间的"对偶性"结论，这不是偶然的。我们还会看到，实际上不含回路的子图聚合和不含割集的子图聚合都具有拟阵结构，而对偶性正是这个事实的结论。还要研究可用于网络分析的"着色"定理和弧染色引理。最后讨论"Greedy"算法，它是寻找加权连通图中最小费用生成树的Kruskal算法的推广。

10.1 基本定义

描述拟阵有若干个等价公理系统，我们开始讨论所谓独立性公理。在10.3节中，将推导出一些等价公理系统。

拟阵M是有限集S和S的子集聚合$\mathscr{I}$，从而满足下列称为"独立性公理"的公理：

I-1　$\varnothing\in\mathscr{I}$。

I-2.如果$X\in\mathscr{I}$及$Y\subseteq X$，那么$Y\in\mathscr{I}$。

I-3如果X和Y是$\mathscr{I}$的元，且$|X|=|Y|+1$，则存在$x\in X-Y$，并使得$Y\cup x\in\mathscr{I}$。

S的元素叫做拟阵M的元素，$\mathscr{I}$的元叫做拟阵M的独立集，M的最大独立集叫做M的基。M的基的聚合用$\mathscr{B}(M)$表示，或简记为$\mathscr{B}$。

S的不属于$\mathscr{I}$的子集被称为是相关的。S的最小相关子集叫做M的回路。如果$\{x\}$是相关的，则S的元素x叫做M的环。M的回路的聚合用$\mathscr{C}(M)$表示，或简记为$\mathscr{C}$。

M的秩函数ρ给S的每个子集A指定一个非负整数$\rho(A)$，其定义为

$$\rho(A)=\max\{|X|:X\subseteq A,X\in\mathscr{I}\}$$

$\rho(A)$叫做A的秩，拟阵M的秩是集合S的秩，记为$\rho(M)$。

现在举几个例子。

令S是矢量空间的有限子集。如前所见，S中所有线性独立矢量的子集族满足独立性公理I-1到I-3，于是，S的这些子集构成了S上的拟阵的独立集聚合。在这个拟阵中，S的子集X的秩等于由X所生成的矢量空间的维数。

设G是具有边集E的无向图，我们可以定义E上的两个拟阵，

先考虑E的所有不含回路的子集聚合$\mathscr{I}$。很显然，$\mathscr{I}$满足I-1和I-2。我们容易证明：$\mathscr{I}$满足I-3（练习题4.8），于是$\mathscr{I}$是E上的拟阵M的独立集聚合。M的每一个基都是G的一个生成林。在这个拟阵中，E的任何子集X的秩等于G中由X所导出的子图的秩。进而，M的每个回路也是G的一个回路。由于这个原因，M就叫做G的回路拟阵。

再考虑不包含G的任何割集的E的所有子集族$\mathscr{I}^*$。可以证明$\mathscr{I}$满足公理I-1到I-3（练习题4.9），所以它是E上的拟阵M^*的独立集族。M^*的每一个基都是G的一个补生成林。在这个拟阵中，E的任何子集X的秩等于G的由X所导出的子图的零度。进而，M^*的每个回路是G的一个割集，拟阵M^*叫做G的割集拟阵或链束拟阵。

上面定义的两个拟阵M和M^*有个有趣的性质，那就是，一个拟阵的基是另一个拟阵基在E中的补集。这个结论对有限集合S（并非必须是一个图的边集）上的任何拟阵都是成立的。也就是说，对于集合S上的任何拟阵M_1，都存在一个S上的拟阵M^*，而M^*的基是M的基的补集。这个结论在10.4节讨论。

拟阵的另一个例子是定义在图的顶点集上的匹配拟阵。

定理10.1. 令G是具有顶点集V的无向图，$\mathscr{I}$是全部子集$I \subseteq V$的聚合，使得I中元素在G的某个匹配中被饱和。那么$\mathscr{I}$就是V上拟阵的独立集的聚合。

证明

$\mathscr{I}$显然满足公理I-1和I-2。

为了证明$\mathscr{I}$满足公理I-3，考察$\mathscr{I}$的两个分别包含p和$p+1$个顶点的元I_p和I_{p+1}，并令X_p和X_{p+1}分别是饱和I_p和I_{p+1}个元素的任意两个匹配。那么会出现两种情况：

情况1　假定某个元素$x \in I_{p+1} - I_p$在X_p中是饱和的，那么X_p就使$I_p \cup x$饱和，并且满足公理I-3。

情况2　假定$I_{p+1} - I_p$中没有元素在X_p中被饱和，那么考虑边集$X_p \oplus X_{p+1} = (X_p - X_{p+1}) \cup (X_{p+1} - X_p)$上的子图$G'$。由定理8.19可知，$G'$的每一个片或者是

1.边交替出现在X_p和X_{p+1}中的回路

或者是

2.边交替出现在X_p和X_{p+1}中的路径，并且它的端点在两个匹配的某一个中是非饱和的。

因为$|I_{p+1} - I_p| > |I_p - I_{p+1}|$，所以$G'$中存在一条从$v \in I_{p+1} - I_p$到不属于$I_p - I_{p+1}$的顶点的路径$P$。而$X_p \oplus P$是饱和$v$和$I_p$中所有元素的匹配。所以$I_p \cup v$是$\mathscr{I}$的元，则$\mathscr{I}$满足公理I-3。

举例，参看图10.1(a)所示图G。集合$I_4 = \{v_1, v_2, v_3, v_4\}$和$I_6 = \{v_1, v_2, v_4, v_5, v_6\}$分别在匹配$X_4 = \{e_1, e_2\}$和$X_6 = \{e_2, e_3, e_4\}$中是饱和的。边集$X_4 \oplus X_6 = \{e_1, e_3, e_4\}$上的子图示于图10.1($b$)。这是一条从顶点$v_6 \in I_6 - I_4$到顶点$v_6 \notin I_4 - I_6$的路径$P$。匹配$X_4 \oplus P = \{e_1, e_3, e_4\}$饱和3集合$I_4 \cup v_6$。

如果在保持独立性的S_1和S_2的元素之间存在一一对应关系，则集合S_1和S_2上的两个拟阵M_1和M_2，就是同构的。

如果S上的拟阵与图G的回路拟阵同构，则S上的拟阵M称为图示的。如果M与图G的

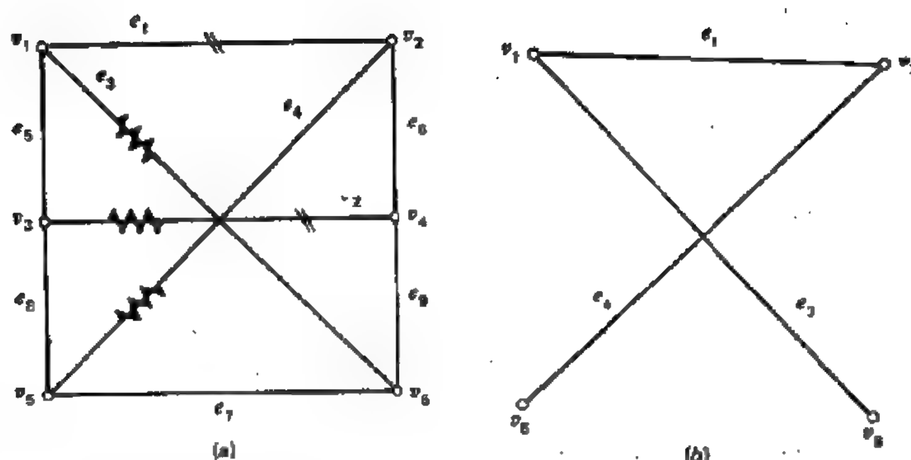


图 10.1 (a)标明匹配x_1和x_2的图；(b)在边集$x_1 \oplus x_2$上的G的子图。

割集拟阵同构，则M称为补图示的。

10.2 基本性质

本节将建立拟阵的几个基本性质。所有这些性质，在图论中都已经提到过。

首先考虑拟阵M的独立集和基。

定理10.2（增广定理） 令X和Y是拟阵M的任意两个独立子集。如果$|X|<|Y|$，那么存在$Z \subseteq Y-X$，并使$|X \cup Z|=|Y|$，且$X \cup Z$在M中是独立的。

证明

令Z_0是符合下列条件的集合：

1. $Z_0 \subseteq Y-X$；

2. $X \cup Z_0$在M中是独立的；

以及

3. 如果对于$Z \subseteq Y-X$，$X \cup Z$在M中是独立的，那么$|X \cup Z_0| \geqslant |X \cup Z|$。

假定$|X \cup Z_0|<|Y|$。那么存在一个使$|Y_0|=|X \cup Z_0|+1$的集合$Y_0 \subseteq Y_0$。因为Y_0在M中是独立的，所以由公理I-3可知，存在一个元素$y \in Y_0-(X \cup Z_0|$，以使$X \cup Z_0 \cup y$在M中是独立的。集合$Z_0 \cup y$与Z_0的选择相矛盾，于是$|X \cup Z_0| \geqslant |Y|$，定理得证。 □

推论10.2.1 集合S上的拟阵M的所有基具有相同的基数，并等于M的秩。

证明

若结论不成立，令B_1、B_2是基，且$|B_1|>|B_2|$。那么由增广定理可知，存在$Z \subseteq B_2-B_1$，以使$B_1 \cup Z$是独立的。但是，这与B_1在$\mathscr{I}$中是最大独立集相矛盾。 □

下面是上述结论的推广，也可用同样的增广定理加以证明。

推论10.2.2 令M是集合S上的拟阵，且$A \subset S$。那么A的所有最大独立子集都具有相同的基数。 □

下面叙述定理10.2的另一个推论，证明略去。

推论10.2.3 如果B_1和B_2是拟阵M的基，且$x \in B_1-B_2$，那么存在$y \in B_2-B_1$，以使$(B_1 \cup y)-x$是M的基。 □

下面介绍拟阵秩函数的某些性质。

定理10.3. 集合S上的拟阵M的秩函数ρ满足下列性质：

1. $0\leqslant\rho(A)\leqslant|A|$。
2. 如果$A\subseteq B$，那么$\rho(A)\leqslant\rho(B)$。
3. $\rho(A)\leqslant\rho(A\cup x)\leqslant\rho(A)+1$。
4. 子模不等式

$$\rho(A)+\rho(B)\geqslant\rho(A\cup B)+\rho(A\cap B).$$

5. 如果$\rho(A\cup y)=\rho(A\cup x)=\rho(A)$，那么$\rho(A\cup x\cup y)=\rho(A)$。

证明

前三条性质很容易从秩函数的定义推出。

为了证明性质4，令X是$A\cap B$的最大独立子集。由增广定理可知，$A\cup B$中存在包含X的最大独立子集Y。令$Y=X\cup V\cup W$，这里$V\subseteq A-B$且$W\subseteq B-A$。因为$X\cup V$是A的独立子集，且$X\cup W$是B的独立子集，则有

$$\rho(A)\geqslant|X\cup V|$$

以及

$$\rho(B)\geqslant|X\cup W|$$

于是

$$\begin{aligned}\rho(A)+\rho(B)&\geqslant|X\cup V|+|X\cup W|\\&=2|X|+|V|+|W|\\&=|X|+(|X|+|V|+|W|)\\&=\rho(A\cap B)+\rho(A\cup B)\end{aligned}$$

性质5是性质4的直接结果。 □

接下来讨论拟阵的回路。下面叙述直接从回路定义得出的某些性质。

1. 回路的每个真子集都是独立的。于是，若C_1和C_2和相异回路，则$C_1\nsubseteq C_2$。
2. 若C是一回路，则$\rho(C)=|C|-1$。
3. 集合S上的拟阵M不含有回路，当且仅当S的所有子集都是独立的。即S是拟阵M的唯一基。

定理10.4 如果C_1和C_2是拟阵M的两个不同回路，且$x\in C_1\cap C_2$，那么存在回路C_3，并使$C_3\in(C_1\cup C_2)-x$。

证明

令$C'=(C_1\cup C_2)-x$，只要证明C'是相关的，即$\rho(C')<|C'|$，则定理得证。

因为C_1和C_2是相异回路，所以$C_1\cap C_2$是C_1和C_2两者的真子集。于是$C_1\cap C_2$是独立的。故

$$\rho(C_1\cap C_2)=|C_1\cap C_2|$$

进而，

$$\rho(C_1)=|C_1|-1$$

以及
$$\rho(C_2)=|C_2|-1$$

利用上述结论和秩函数的子模性（定理10.3），可得

$$\begin{aligned}\rho(C_1\cup C_2)&\leqslant\rho(C_1)+\rho(C_2)-\rho(C_1\cap C_2)\\&=|C_1|-1+|C_2|-1-|C_1\cap C_2|\\&=|C_1\cup C_2|-2\\&<|C'|\end{aligned}\tag{10.1}$$

因为$C'\subseteq C_1\cup C_2$，又得

$$\rho(C')\leqslant\rho(C_1\cup C_2)\tag{10.2}$$

联立式（10.1）和式（10.2），则有

$$\rho(C')<|C'|$$ □

推论10.4.1 如果A是集合S上拟阵M的独立集，那么对于任意$x\subseteq S$，$A\cup x$至多包含一个回路。

证明

设对于某个$x\in S$，存在两个不同回路C_1和C_2，并使$x\in C_1\cap C_2$，$C_1\subseteq A\cup x$，及$C_2\subseteq A\cup x$。那么根据前面定理可知，回路$C'\subseteq(C_1\cup C_2)-x$存在。于是，$(C_1\cup C_2)-x$是非独立的。但是$(C_1\cup C_2)-x\subseteq A$，这与$A$的独立性相矛盾。□

推论10.4.2 如果B是集合S上拟阵M的基，且$x\in S-B$，那么存在唯一回路$C=C(x,B)$，以使$x\in C\subseteq B\cup x$。

证明

因为B是最大独立集，所以$B\cup x$包含回路，由前面的推论，可知这样的回路是唯一的。□

上面推论中定义的回路$C(x, B)$，称为对应于基B的x的基本回路。

下面证明比定理10.4更强有力的结论。

定理10.5 如果C_1和C_2是拟阵的相异回路，且$x\in C_1\cap C_2$，那么对于任一元素$y\in C_1-C_2$，存在回路C，并使

$$y\in C\subseteq(C_1\cup C_2)-x$$

证明

假设C_1、C_2、x、y使定理不成立，且$|C_1\cup C_2|$是所有不满足定理的相异回路对中的最小者。

由定理10.4可知，回路$C_3\subseteq(C_1\cup C_2)-x$存在。由$x$和$y$的选取方式，显然有$y\notin C_3$。进而，$C_3\cap(C_2-C_1)\neq\varnothing$，否则有$C_3\subseteq C_1$，这与$C_1$是最小相关集这一事实相矛盾。令$z\in C_3\cap(C_2-C_1)$。

对于回路C_2和C_3，则有：

1. $z\in C_2\cap C_3$，
2. $x\in C_2-C_3$，
3. $C_2\cap C_3$是$C_1\cup C_2$的真子集，因为$y\notin C_2\cap C_3$。

由C_1和C_2的选取方式可知，回路C_4存在，并使

$$x\in C_4\subseteq(C_2\cup C_3)-z$$

再来考虑回路C_1和C_4，它们满足：

1.$x \in C_1 \cap C_4$，

2.$y \in C_1 - C_4$，因为$y \notin C_2 \cup C_3$，

3.$C_2 \cup C_4$是$C_1 \cup C_2$的真子集，因为$z \notin C_1 \cup C_4$。

又由C_1和C_2的选取方式可知，回路C_5存在，并使$y \in C_5 \subseteq (C_1 \cup C_4) - x$。

因为$C_1 \cup C_4$是$C_1 \cup C_2$的真子集，所以我们找到了回路C_5，并使$y \in C_5 \subseteq (C_1 \cup C_2) - x$，这与$C_1$和$C_2$的选择相矛盾。 □

10.3 等价公理系统

本节将建立拟阵定义的另外一些公理系统。这些等价描述可以帮助我们加深对拟阵结构的了解。

首先给出独立性公理的等价集合。

定理10.6（独立性公理） 集合S的子集聚合$\mathscr{I}$是S上拟阵的独立集的集合，当且仅当$\mathscr{I}$满足公理I-1，I-2以及

I-3 如果A是S的任一子集，那么A的所有最大子集$Y(Y \in \mathscr{I})$都具有相同的基数。

证明 □

证明留作练习，参见推论10.2.2。

定理10.7（基公理） 令$\mathscr{B}$是拟阵的基的集合，那么

B-1 $\mathscr{B} \neq \varnothing$，$\mathscr{B}$中不存在以其它集合为其真子集的集合。

B-2 如果B_1、$B_2 \in \mathscr{B}$，$x \in B_1$，则满在$y \in B_2$，并使$(B_1 - x) \cup y \in \mathscr{B}$。

相反，如果$\mathscr{B}$是集合S的子集聚合，且拟足B-1和B-2，那么

$$\mathscr{I} = \{I \mid I \subseteq B, \text{ 对某个} B \in \mathscr{B}\}$$

就是S上拟阵的独立集聚合。

证明

已经证明拟阵的基满足B-1和B-2。（回忆基的定义和推论10.2.3。）

假定$\mathscr{B}$满足B-1和B-2，现在证明：定理中所定义的$\mathscr{I}$是S上拟阵是独立集聚合。

注意，B-1和B-2意味着对于$x \in B_1 - B_2$，存在$y \in B_2 - B_1$，并使得$(B_1 - x) \cup y$是$\mathscr{B}$的元。

$\mathscr{I}$显然满足公理I-1和I-2。为了证明它也满足公理I-3，先证明$\mathscr{B}$的所有元具有相同的基数。

假定存在B_1，$B_2 \in \mathscr{B}$，且$|B_1| > |B_2|$。那么重复应用B-2，就可从B_2中移去$B_2 - B_1$，并用$A \subset B_1 - B_2$来代替，且$|A| = |B_2 - B_1|$，并使得$B_3 = (B_2 \cup B_1) \cup A$是$\mathscr{B}$的元。但是$B_3 \subset B_1$，这与B-1相矛盾。所以$\mathscr{B}$的所有元具有相同的基数。

再考虑$\mathscr{I}$的任意两个相异元X和Y，且$|Y| = |X| + 1$。令$X \subseteq B_1$，$Y \subseteq B_2$，及B_1，$B_2 \in \mathscr{B}$。令

$$X = \{x_1, x_2, \cdots, x_k\}$$
$$B_1 = \{x_1, x_2, \cdots, x_k, b_1, b_2, \cdots b_k\}$$

$$Y=\{y_1,\ y_2,\ \cdots,\ y_k,\ y_{k+1}\}$$

$$B_2=\{y_1,\ y_2,\ \cdots,\ y_k, y_{k+1},\ c_1,\ c_2,\ \cdots,\ c_{q-1}\}$$

现在来看B_1-b_q。由B-2可知，$z\in B_2$存在，并使$B'=(B_1-b_q)\cup z$是$\mathscr{B}$的元。如果$z\in Y$，那么$(x\cup z)\subseteq B'$，因此它是$\mathscr{B}$的元。所以，在这种情况下$\mathscr{I}$满足I-3。

如果$z\notin Y$，令$B''=B'-b_{q-1}$。又根据B-2可知，$z'\in B_2$存在，并使$B''=(B'-b_{q-1})\cup z'$是$\mathscr{B}$的元。如果$z'\in Y$，那么$X\cup z'\subseteq B''$，于是$(X\cup z')\in\mathscr{I}$，并且满足公理I-3。

如果$z'\in Y$，把b_{q-2}从B''中移去，等等，如此进行下去。因为$|(b_1,\ b_2,\ \cdots,\ b_q)|>|(c_1,\ c_2,\ \cdots,\ c_{q-1})|$，所以最多经过$q$步就可用$Y$的元素来代替某个$b_i$。因此，满足了公理I-3。 □

定理10.8（秩公理） 集合S上拟阵M的秩函数ρ满足：

R-1 $0\leqslant\rho(A)\leqslant|A|$，对所有$A\subseteq S$。

R-2 如果$A\subseteq B\subseteq S$，那么$\rho(A)\leqslant\rho(B)$。

R-3 对于任意A，$B\subseteq S$，

$$\rho(A)+\rho(B)\geqslant\rho(A\cup B)+\rho(A\cap B)$$

相反，如果在有限集S上定义的整值函数ρ满足R-1、R-2以及R-3，那么集合

$$\mathscr{I}=\{I\,|\,\rho(I)=|I|,\ I\subseteq S\}$$

就是S上拟阵的独立集聚合。

证明

在定理10.3中已经证明：拟阵的秩函数满足R-1、R-2和R-3。

为了证明逆定理，假定ρ满足R-1、R-2和R-3。由R-1可知，$\rho(\varnothing)=0$，所以$\varnothing\in\mathscr{I}$，并且满足公理I-1。

令$B\in\mathscr{I}$和$A\subsetneq B$，那么由R-3，则得

$$\begin{aligned}\rho(A)+\rho(B-A)&\geqslant\rho(A\cup(B-A))+\rho(A\cap(B-A))\\&=\rho(B)+\rho(\varnothing)\\&=\rho(B)\end{aligned}\qquad(10.3)$$

如果$\rho(A)<|A|$，那么由R-1，可得

$$\begin{aligned}\rho(A)+\rho(B-A)&<|A|+|B-A|\\&=|B|\end{aligned}$$

这与式（10.3）相矛盾。于是$\rho(A)=|A|$，从而A也是$\mathscr{I}$的元，所以满足公理I-2。

为了证明I-3，注意R-2和R-3隐含了下列特性：

R-3′ 如果$\rho(A\cup x)=\rho(A\cup y)=\rho(A)$，那么$\rho(A\cup x\cup y)=\rho(A)$。

令X和Y是$\mathscr{I}$的两个相异元，且$|X|=k$和$|Y|=k+1$。假定对于每个$y\in Y-X$，$\rho(X\cup y)=k$。那么重复应用R-3′，就可得到

$$\rho(X\cup(Y-X))=\rho(X\cup Y)=k$$

这与$\rho(Y)=k+1$相矛盾。所以必然存在一个具有性质$\rho(X\cup y)=k+1$的元素$y\in Y-X$，从而满足公理I-3。 □

定理 10.9（回路公理） 令$\mathscr{C}$是集合S上拟阵的回路集合，那么

C-1　$\varnothing \notin \mathscr{C}$，以及$\mathscr{C}$中不存在以其它集合为其真子集的集合。

C-2　如果$C_1 C_2 \in \mathscr{C}$，$C_1 \neq C_2$，以及$x \in C_1 \cap C_2$，那么存在C_3，并使

$$C_3 \subseteq (C_1 \cup C_2) - x$$。

相反，如果$\mathscr{C}$是满足C-1和C-2集合S的子集聚合，那么集合$\mathscr{I} = \{I | C \not\subset I$，对于所有$C \in \mathscr{C}\}$是$S$上拟阵的独立集聚合。

证明

前面已经证明，拟阵的回路满足C-1和C-2（见定理10.4）。

已知$\mathscr{C}$满足C-1和C-2，下面证明在定理中所定义的集合$\mathscr{I}$是S上拟阵的独立集聚。合要做到这一点，就得证明$\mathscr{I}$满足公理I-1，I-2和I-3'（见定理10.6）。

注意：$\mathscr{I}$是S中不含$\mathscr{C}$的任何元的所有子集的聚合。因而$\mathscr{I}$满足公理I-1和I-2。

为了证明$\mathscr{I}$满足I-3'，考虑S的任意子集。令S_1和S_2是属于$\mathscr{I}$的相异最大子集。假定$|S_2| > |S_1|$，下面将得出矛盾。

因为S_1和S_2是相异的，所以$S_1 - S_2 \neq \varnothing$和$S_2 - S_1 \neq \phi$。令$x \in S_2 - S_1$那么显然$S_1 \cup x \notin \mathscr{I}$。于是有$C \in \mathscr{C}$，且$x \in C \subseteq S_1 \cup x$。进而，$C \cap (S_1 - S_2) \neq \varnothing$，否则$C \subset S_2$，这与$S_2 \in \mathscr{I}$相矛盾。

然后令$x' \in C \cap (S_1 - S_2)$和$S_3 = (S_1 - x') \cup x$。注意$|S_3| = |S_1|$。首先证明：

1.$S_3 \in \mathscr{I}$。

显然$S_1 - x' \in \mathscr{I}$。若$S_3 \notin \mathscr{I}$，则有$C' \in \mathscr{C}$，且$x \in C' \subseteq S_3 \subset S_1 \cup x$。进而$C \neq C'$，因为$x \notin C'$。故$C$和$C'$是$\mathscr{C}$的相异元，并使得

（a）$x \in C \cap C'$。

（b）$C \subset S_1 \cup x$和$C' \subset S_1 \subset x$。

那么由C-2可知，$C'' \in \mathscr{C}$存在，且有$C'' \subseteq (C \cup C') - x \subseteq S_1 \in \mathscr{I}$。这与$\mathscr{I}$的元不包含$\mathscr{C}$的元相矛盾。于是$S_3 \in \mathscr{I}$。

接下来证明：

2.$S_3 \in \mathscr{I}$是A的最大子集。

如果不是，令$S' \in \mathscr{I}$是A的最大子集，且$S_3 \subset S'$。因$x' \notin S'$，否则$S_1 \subset S'$，这与S_1在A中是最大者相矛盾。因此$S' \cup x' \notin \mathscr{I}$。于是存在$C'' \in \mathscr{C}$，且有$x' \in C'' \subseteq S' \cup x'$，并且$C'' \cap (S' - S_1) \neq \varnothing$，否则$C'' \subseteq S_1$。令$x'' \in C'' \cap (S' - S_1)$，那么可以证明（如同1的证明）$(S' - x'') \cup x'' \in \mathscr{I}$。但是$S_1 \subset (S' - x'') \cup x' \in \mathscr{I}$，这与$S_1$在$A$中是最大者相矛盾，所以$S_3$是$A$的最大子集。

注意：我们是通过在S_1中用元素$x \in S_2 - S_1$代替$x' \in S_1 - S_2$来构造S_3的。因为$|S_2| > |S_1|$，所以只要重复有限次这种构造，就可以得到A的最大子集S_n，且有$S_n \in \mathscr{I}$和$S_n \subset S_2$，这与$\rho(Y) = k+1$相矛盾。所以存在元素$y \in Y - X$，且有$\rho(X \cup y) = k+1$的性质，于是$\mathscr{I}$满足公理I-3。□

10.4　拟阵的对偶性和拟图

本节首先介绍拟阵对偶性的概念，然后讨论拟阵及其对偶的一些结论。接着证明“若

色”定理，并给出由此定理所产生的“拟图”的定义。最后在拟图基础上以推导拟阵的Minty自-对偶公理系统的来结束本节。

开始先介绍Whitney文献[10.1]的定理，对拟阵的对偶下定义。

定理 10.10 令$\mathscr{B}$是集合S上的拟阵M的基的集合。那么$\mathscr{B}^*=\{S-B \mid B\in\mathscr{B}\}$是$S$上拟阵$M^*$的基的集合。

证明

显然，$\mathscr{B}^*$满足公理B-1。为了证明它也满足B-2，考虑$\mathscr{B}$的任意两个元B_1^*和B_2^*，且$B_1^*=S-B_1$和$B_2^*=S-B_2$。令$x\in B_1^*-B_2^*$，那么$x\in B_2-B_1$。由习题10.4可知，$y\in B_1-B_2$存在，并使得$B_3=(B_1-y)\cup x$是M的基。现在，$y\in B_2^*-B_1^*$和$(B_1^*-x)\cup y=S-((B_1-y)\cup x)=S-B_3=B_3^*$，于是$\mathscr{B}^*$满足公理B-2。所以$\mathscr{B}^*$是$S$上拟阵$M$的基。 □

定理中所定义的M^*称为M的对偶拟阵或简称为对偶。很容易看出：M也是M^*的对偶。所以，我们把M和M^*互称为对偶拟阵。

10.1节中所定义图的回路拟阵和割集拟阵现在都可以看作对偶拟阵。

M^*的基称为M的补基。类似地，M^*的回路称为M的补回路，M^*的环称为M的补环，M^*的秩函数称为M的补秩函数，等等。M^*的秩函数用ρ^*表示。如果B是M的基，那么补基$S-B$就用B^*表示。M的补基聚合记为$\mathscr{B}^*(M)$，补回路聚合记为$\mathscr{C}^*(M)$。

现在推导有关拟阵及其对偶的一些结论。其中有许多结论在图论中是很熟悉的（见第二章）。

由对偶的定义，显然有

$$\rho^*(M)=|S|-\rho(M) \tag{10.4}$$

下面的定理扩展了ρ和ρ^*之间的关系。

定理 10.11. 如果M和M^*是集合S上的对偶拟阵，那么对于任何$A\subseteq S$，

$$\rho^*(A)=|A|+\rho(S-A)-\rho(M) \tag{10.5}$$

证明

考虑$A\subseteq S$，令B^*是M^*的基，并使得$|B^*\cap A|$为最大。那么B是M的基，且$|B\cap(S-A)|$为最大。由秩函数的定义可知，

$$\rho^*(A)=|B^*\cap A| \tag{10.6}$$

以及

$$\rho(S-A)=|B\cap(S-A)| \tag{10.7}$$

又

$$|B^*\cap A|=|A|-|B\cap A|$$

以及

$$\begin{aligned}|B\cap(S-A)|&=|B|-|B\cap A|\\&=\rho(M)-|B\cap A|\end{aligned}$$

所以由式（10.6）和（10.7），可得

$$\rho^*(A)=|A|+\rho(S-A)-\rho(M)$$ □

注意：对于拟阵的回路、基等的每一条陈述，都有一条涉及到该拟阵的补回路、补基

等的对偶陈述。这是因为补回路、补基等本身就是拟阵的回路、基等等。

在本节所有的引理和定理中，如果可用，也包括对偶陈述。这些对偶陈述的证明以简明方式给出。

引理 10.1. 令M是集合S上的拟阵。如果$A\subseteq S$在M中是独立的，那么$S-A$就包含了M的补基。对偶地，如果$A^*\subseteq S$在M^*中是独立的，那么$S-A^*$就包含了M的基。

证明

M中有基B，并满足$A\subseteq B$，所以$S-A$包含对应的补基B^*。 □

定理 10.12. 令M是集合S上的拟阵。如果A和A^*是S的子集，且有$A\cap A^*=\varnothing$，A在M中是独立的，A^*在M^*中也是独立的，那么存在M的基B，且有$A\subseteq B$，$A^*\subseteq B^*$。

证明

根据前面的引理，$S-A^*$包含了M的基。因为$A\subseteq S-A^*$在M中是独立的，所以根据定理10.2可知，在M中有基B，且$A\subseteq B\subseteq S-A^*$，因而$A^*\subseteq B^*$。 □

引理 10.2 令M是集合S上的拟阵，则M的每个基B与M的任何补回路有非零交。

对偶地，M的每个补基与M的任何回路有非零交。

证明

如果，对于M的任何补回路C^*，有$C^*\cap B=\varnothing$，那么$B^*=S-B$将包含C^*，这与B^*在M中是独立的相矛盾。

令B^*是集合S上拟阵的任何补基。由推论10.4.2的对偶结论可知，对于任意$x\in B$，$B^*\cup x$恰好包括了一个补回路。这个补回路称为x相对于B的基本补回路。注意，这个补回路恰好包含B的一个元素，即元素x。

引理 10.3. 令M是一拟阵，对于M的任何独立集A，则存在一补回路，它恰好含有A的一个元素，进而，如果$|A|<\rho(M)$，则存在一补回路，它与A有零交。

对偶地，对于M^*的任何独立集A^*，则存在一回路，它恰好含有A^*的一个元素。进而，若$|A^*|<\rho(M^*)$，则存在一回路，它与A^*有零交。

证明

令B是M的基，且$A\subseteq B$又令C^*是元素$x\in B$对应于基B的基本补回路。如果$x\in A$，则$C^*\cap A=\{x\}$；如果$x\in B-A$，则$C^*\cap A=\varnothing$。从而引理得证。 □

定理 10.13. 令M是集S上的拟阵。S的子集X是M的基，当且仅当X是与M的每个补回路有非零交的最小子集。

对偶地，S的子集X^*是M的补基，当且仅当它是与M的每个回路有非零交的最小子集。

证明

必要性很容易从引理10.2和10.3得出。

为了证明充分性，令X是与M的每个补回路有非零交的S的最小子集。那么$S-X$是不包含M的补回路的最大子集。所以，由定义，$S-X$是M的补基，从而X是M的基。 □

下面将得到回路和补回路的一些新特性。

定理 10.14. 令M是集合S上的拟阵。S的子集是M的回路，当且仅当它是与M的每

个补基有非零交的最小子集。

对偶地，S的子集X^*是M的补回路，当且仅当它是与M的每个基有非零交的最小子集。

证明

见定理2.13的证明。 □

令C^*是S上拟阵M的补回路。根据前面的定理可知，C^*是与M的每个基有非零交的最小子集。所以$S-C^*$是不含M的基的最大子集。于是有下述引理。

引理 10.4 如果C^*是集合S上拟阵M的补回路，那么对任一$x\in C^*$，$(S-C^*)\cup x$包含了M的基B。

对偶地，如果C是集合S上拟阵M的回路，那么对任一$x\in C$，$(S-C)\cup x$包含了M的补基。 □

定理10.15 令M是集合S上的拟阵。S的子集X是M的回路，当且仅当它是一最小子集，且对于M的每个补回路C^*，有$|X\cap C^*|\neq 1$。

对偶地，S的子集X^*是M的补回路，当且仅当它是一最小子集，且对于M的每个回路C，有$|X^*\cap C|\neq 1$。

证明

必要性　通过引出矛盾加以证明。对于回路C的每个真子集C'，根据定理10.3，则存在补回路C^*，并使$|C^*\cap C'|=1$。

假定对于某个回路C和某个补回路C^*，$|C\cap C^*|=1$，并令$C\cap C^*=\{x\}$，考虑$S'=S-C^*$和$C'=C-x$。很显然，$C'\subseteq S'$。由引理10.4可知，$S'\cup x$包含一个基。然后令$B\subset S'\cup x$是一个基，且$C\subseteq B$。注意$x\in B$，所以B中含有回路$C=C'\cup x$，这就与假定产生矛盾。

充分性　如果X是S的子集，并且对每个补回路C^*，有$|X\cap C^*|\neq 1$，那么X就应包含回路，否则X在M中将是独立的；由引理10.3可知，将存在一个补回路，它恰好包含了X的一个元素。

令C是X中的回路。显然，从定理的必要性部分可知，对于每个补回路C^*，有$|C\cap C^*|=1$。于是$X=C$，否则与X的最小性相矛盾。 □

现在介绍有限集S的涂色概念，然后给出“涂色”定理。

集合S的涂色是把S划分为三个子集R、G和B，且使$|G|=1$。为了形象起见，我们假设把R中元素“涂红”，G中元素“涂绿，B中元素“涂蓝”。

定理10.16（涂色定理） 令M是集合S的拟阵。对于S的任何涂色，或者是

1. M的回路C包含绿元素，但不包含蓝元素；

或者是

2. M的补回路C^*包含绿元素，但不包含红元素。

证明

设绿元素在某个回路中，否则它是补圈（为什么），并且所得定理也就没有多大意义了。

假定对于S的某个涂色，论断1不成立。那么考虑子集$S'=R\cup G$。显然，S'所包含的回路仅由红元素构成。令R'是红元素的最小子集，且使集$S''=S'-R'$不含回路，并令B是一个基，且$S''\subseteq B$，那么容易看出：对应于B的任一元素$y\in R'$的基本回路仅由红色元素构成。

令C^*是绿元素对应于B的基本补回路。假设C^*含有一个红元素x，则$x\in R'$以及$|C^*\cap C_x|=1$，这里C_x是x对应于B的基本回路。这与定理10.15相矛盾，因此C^*是一个包含绿元素而不包含红元素的补回路。从而论断2成立。 □

定理10.15和10.16引出了Minty文献[10.2]的拟图定义。

拟图是一个由有限集S和S的两个非空子集聚合$\mathscr{C}$和$\mathscr{D}$组成的结构$(S,\ \mathscr{C},\ \mathscr{D})$，其中$\mathscr{C}$和$\mathscr{D}$满足下列要求：

G-1. 如果$C\in\mathscr{C}$和$D\in\mathscr{D}$，则$|C\cap D|\neq 1$。

G-2. 对于S的任何涂色，或者是

  a. $\mathscr{C}$的元包含绿元素而不包含蓝元素；或者是

  b. $\mathscr{D}$的元包含绿元素而不包含红元素。

G-3. $\mathscr{C}$中不包含以其它元为自己的真子集的元；

  $\mathscr{D}$中不包含以其它元为自己的真子集的元。

定理10.17 令M是集合S的拟阵，则$(S,\ \mathscr{C}(M),\ \mathscr{C}^*(M))$是一个拟图。

证明

可以从定理10.15和10.16导出结论。 □

现在来论证上面定理的逆定理。

定理10.18 令$(S,\ \mathscr{C},\ \mathscr{D})$是一个拟图。那么$B\subseteq S$是不含$\mathscr{C}$的元的最大子集，当且仅当$B$不包含$\mathscr{C}$的元和$S-B$不包含$\mathscr{D}$的元。

证明

必要性 令B是S不包含$\mathscr{C}$的元的最大子集。假设$S-B$包含了$\mathscr{D}$的元D。令$x\in D$，那么$B\cup x$包含了$\mathscr{C}$的元C，且$x\in C$。所以$C\cap D=\{x\}$，这与G-1相矛盾。故$S-B$不包含$\mathscr{D}$的元。

充分性 令$B\subseteq S$不包含$\mathscr{C}$的元，$S-B$不包含$\mathscr{D}$的元。现在证明，对于任何$x\in S-B$，$B\cup x$包含了$\mathscr{C}$的元。

考虑S的涂色，其中B的元素是红的，x的元素是绿的，$S-B$的其余元素是蓝的。显然，$\mathscr{D}$中不存在只包含绿元素和蓝元素的元，故由G-2，$B\cup x$包含$\mathscr{C}$的元。 □

定理10.19 令$(S,\mathscr{C},\mathscr{D})$是一个拟图。如果$B$是$S$不包含$\mathscr{C}$的元的最大子集，则$S-B$是$S$不包含$\mathscr{D}$的元素的最大子集。

证明

可以从定理10.18及其对偶定理推出。 □

定理10.20 令$(S,\ \mathscr{C},\ \mathscr{D})$是一个拟图。那么$\mathscr{C}$是$S$上拟阵$M$的回路聚合，$\mathscr{D}$是$M$的补回路聚合。

证明

首先说明$\mathscr{C}$是S上拟阵M_1的回路集。

$\mathscr{C}$显然满足公理C-1。

为了证明$\mathscr{C}$也满足C-2，令C_1和C_2是$\mathscr{C}$的任意两个相异元，且$x\in C_1\cap C_2$和$y\in C_1-C_2$。设在S的某个涂色中，y是绿的，x是蓝的，$C_1\cup C_2$的其余元素是红的，S的其余元素是蓝的（见图10.2）。

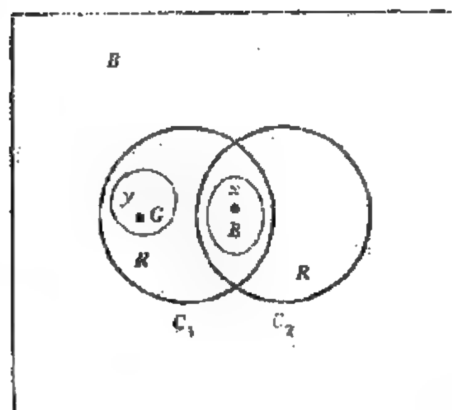


图 10.2

$\mathscr{D}$中不存在包含绿元素而又不包含红元素的元D。如若$\mathscr{D}$中存在这样的元素D，那么就可以证明：或者$|C_1\cap D|=1$或者$|C_2\cap D|=1$，因而与G-1相矛盾。所以由G-2可知，$\mathscr{C}$中存在包含绿元素而不包含蓝元素的元C_3。由G-3又知，$\{y\}$不是$\mathscr{C}$的元，故可推得$y\in C_3\subset(C_1\cup C_2)-x$。于是$\mathscr{C}$满足了公理C-2，$\mathscr{C}$是S上拟阵$M_1$的回路集合。

同法可证：$\mathscr{D}$是S上拟阵M_2的回路集合。

由定理10.19可知，M_1的基是M_2的基的补，所以M_1和M_2是对偶拟阵。 □

拟图公理系统和回路公理系统的等价性可由定理10.17和10.20推出。Minty在文献[10.2]中给出了这个结论，并对以拟图为基础的拟阵理论作了精辟的阐述。

10.5 约束、收缩和拟阵的子式

考察具有边集E的图G。对于E的任何子集T，可以定义两个相关的图，记为$G|T$和$G\cdot T$。图$G|T$称为G对T的约束，它是从G除去（开路）属于$E-T$中的边，然后除去由此所产生的孤立顶点所得到的。实际上，$G|T$就是在T上G的导出子图。图$G\cdot T$称为G对T的收缩，它是通过收缩$E-T$中的边所得到的。图10.3示出了图G和对应于$T=\{e_1, e_6, e_7, e_8\}$的图$G|T$和$G\cdot T$。

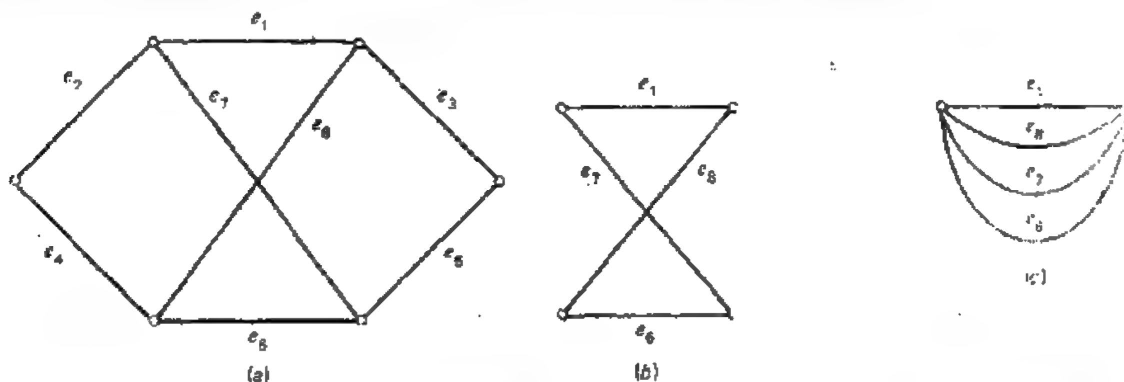


图 10.3 （a）图G；（b）$G|T$；G对T的约束，$T=\{e_1, e_6, e_7, e_8\}$；（c）$G\cdot T$，G对T的收缩。

现在容易证明下列性质：

1. 边的子集X是$G|T$的无圈子图，当且仅当它是G的无圈子图。

2. 边的子集X是$G\cdot T$的无圈子图，当且仅当存在$E-T$的子集Y，并使得Y是$G|(E-T)$的生成林，$X\cup Y$是G的无圈子图。

上述概念促使我们引入了在拟阵上由拟阵元素的子集导出的两个子拟阵。

如果$\mathscr{I}(M)$是集合S上拟阵M的独立集的集合，且$T\subseteq S$，令

$$\mathscr{I}(M|T)=\{X|X\subseteq T,\ x\subseteq\mathscr{I}(M)\},$$

可以证明：$\mathscr{I}(M|T)$是拟阵在T上的独立集的集合。这个拟阵记为$M|T$，并称为M对T的约束。

下面的论断是显而易见的：

1.$X\subseteq T$在$M|T$中是独立的，当且仅当X在M中是独立的。

2.$X\subseteq T$是$M|T$的回路，当且仅当它是M的回路。

3.如果λ是$M|T$的秩函数，那么对于任何$X\subseteq T$，

$$\lambda(X)=\rho(X) \tag{10.8}$$

4.如果$M(G)$是具有边集E的图G的回路拟阵，那么对于任何$T\subseteq E$，

$$M(G)|T=M(G|T)$$

下面定义子拟阵$M\cdot T$。令$\mathscr{I}(M\cdot T)$是T的所有那些子集X的集合，并使得$S-T$存在最大子集Y，且$X\cup Y=\mathscr{I}(M)$。

定理10.21 $\mathscr{I}(M\cdot T)$是T上拟阵的独立集聚合。

证明

令X_1和X_2是$\mathscr{I}(M\cdot T)$的两个元，并使得$|X_2|=|X_1|+1$。那么则存在$S-T$的最大独立子集Y_1和Y_2，以及在M中独立的$X'=X_1\cup Y_1$和$X''=X_2\cup Y_2$。现在$|X''|=|X'|+1$，由公理I-3可知，元素$x\in X''-X'$存在，并使得$X'\cup x$在M中是独立的。注意：$X''-X'=(X_2-X_1)\cup(Y_2-Y_1)$。进而，$x\notin Y_2-Y_1$，因为$Y_1$是$S-T$的最大独立集。所以$x\in X_2-X_1$和$(X_1\cup X_2)\subseteq T$是$\mathscr{I}(M\cdot T)$的元。于是$\mathscr{I}(M\cdot T)$满足公理I-3。还容易看出，它也满足I-2和I-3。 □

上述定理中所定义的拟阵称为M对T的收缩，并记为$M\cdot T$。注意，如果$M(G)$是具有边集E的图G的回路拟阵，那么对于任何$T\subseteq E$，$M(G\cdot T)=M(G)\cdot T$。

令ρ^T是$M\cdot T$的秩函数。那么由$M\cdot T$的定义可知，对于任何$A\subseteq T$，有

$$\rho^T(A)=\rho(A\cup(S-T))-\rho(S-T) \tag{10.9}$$

在第七章，我们已看到了图的一条边的开路与去除和一条边的收缩是对偶运算。特别地，如果G和G'是对偶图，T和T'分别是G和G'对应的边子集，那么$G|T$和$G\cdot T$也是对偶图（见定理7.10）。

下面的定理是用拟阵来模拟这种关系。

定理 10.22 如果M是S上的拟阵及$T\subseteq S$，那么

1.$(M|T)^*=M^*\cdot T$。

2.$(M\cdot T)^*=M^*|T$。

证明

1.令λ是$M|T$的秩函数，λ^*是$(M|T)^*$的秩函数。那么由式（10.5）得知，对于任何$X\subseteq T$，有$\lambda^*(X)=|X|-\lambda(T)+\rho(T-X)$。

令ρ^T是$M\cdot T$的秩函数，$(\rho^*)^T$是$M^*\cdot T$的秩函数。那么由式（10.9）和(10.5)，可得

$$
\begin{aligned}
(\rho^*)^T(X) &= \rho^*(X \cup (S-T)) - \rho^*(S-T) \\
&= \rho^*(S-(T-X)) - \rho^*(S-T) \\
&= |S|-|T|+|X|-\rho(S)+\rho(T-X)-(|S|-|T|-\rho(S)+\rho(T)) \\
&= |X|-\rho(T)+\rho(T-X) \\
&= |X|-\lambda(T)+\lambda(T-X)
\end{aligned}
$$

由式（10.8）可得

$$(\rho^*)^T(X)=\lambda^*(X)$$

因为$(M|T)^*$和$M^*\cdot T$具有相同的秩函数，故有

$$(M|T)^*=M^*\cdot T$$

2.把1中的M^*用M置换，并取其对偶，即得2。 □

如果M是S上的拟阵，且$T\subseteq S$，若N是通过连续对M进行约束和收缩所得到的，则T上的拟阵N称为M的子式，这个专题已由Tutte在文献[10.3]中作了深入的研究。

利用已经介绍的术语，我们可以用拟阵术语表述许多与平面图和对偶图有关的结论。例如，定理7.5可表述如下：

一个图是平面的，当且仅当该图的回路拟阵$M(G)$不包含拟阵$M(K_5)$和$M(K_{3,3})$中任一个为其子式。

10.6 拟阵的可表达性

令M是集合S上的拟阵，$B=\{b_1,b_2,\cdots,b_r\}$是M的基，$B^*=S-B=\{e_1, e_2, \cdots, e_q\}$是$M$的对应的补基。曾记得，$B^*\cap C(e_j)=e_j$，$j=1,\cdots,q$，这里$C(e_j)$是$e_j$的基本回路。类似地，$B\cap C^*(b_j)=b_j$，$j=1, \cdots, r$，这里$C^*(b_j)$是$b_j$的基本补回路。在无向图的情况下（第六章），我们可以定义M对应基B的基本补回路矩阵D_f和基本回路矩阵D_f^*。显然，这两个矩阵具有形式

$$[U \quad D]$$

这里U是单位矩阵，它的列对应于B的元素。具有这种形式的矩阵叫做M对应基B的正则表示。

考虑图G，假定我们给G中的边任意指定方向，并令G'是所生成的有向图。那么由定理6.9可知，G'的关联矩阵是回路拟阵$M(G)$在任何域F上的表示。还可以看出（定理6.10），G'的任何割集矩阵是$M(G)$在任何域F上的正则表示。进而，由定理6.11可知，G'的任何基本回路矩阵是割集拟阵$M^*(G)$在任何域F上的正则表示，故有如下定理。

定理 10.23 图G的回路拟阵和割集拟阵在任何域F上都是可表达的。 □

下面证明本节的主要定理。

定理 10.24 如果S上的拟阵M在任何域F上都是可表达的，那么S上的对偶拟阵也是可表达的。

证明

设M的秩为r，并含有n个元素。令$r\times n$矩阵A是M在F上的一种表示。

令X是所有使$Ax=0$的那些列矢量x的集合。X称为A的零空间，众所周知，在线性

代数中，X的维数等于$n-r$。从X中选出一组$n-r$个线性独立列矢量，并用这些列矢量作为列来构成$n\times(n-r)$矩阵B。注意：$AB=0$。

现在证明：B^t是对偶拟阵M^*在F上的一种表示。为此只要证明A的任何r列是线性独立的，当且仅当B^t的$n-r$列补集也是线性独立的。为此目的，取A的前r列来讨论，这样显然不失一般性。

A的前r列是线性独立的，当且仅当存在属于X的非零列矢量$x=[x_1, x_2, \cdots, x_r, 0, 0, \cdots, 0]^t$，这个矢量$x\in X$存在，当且仅当存在含有$n-r$个元素的列矢量$y\neq 0$，且使

$$x=By$$

现在把B写成

$$\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$$

这里B_1是$r\times(n-r)$矩阵，B_2是$(n-r)\times(n-r)$矩阵，从上面的方程可以看出：$B_2y=0$。因为$y\neq 0$，则可知B_2是奇异的。所以B_2的行，也就是B^t的最后$n-r$行是相关的，从而定理得证。 □

下面的结论是上述定理的简单推论。

推论 10.24.1 令M是集合$S=\{s_1, s_2, \cdots, s_n\}$上秩为$r$的拟阵。如果$M$有正则表示

$$\begin{matrix} s_1 s_2 \cdots s_r & s_{r+1} \cdots s_n \end{matrix}$$
$$[\ U_r \mid A\ ]$$

那么M^*就有正则表示

$$\begin{matrix} s_1 s_2 \cdots s_r & s_{r+1} \cdots s_n \end{matrix}$$
$$[\ -A^t \mid U_{n-r}\ ]$$

这里U_k是$k\times k$单位矩阵。 □

连通图的基本回路和基本割集矩阵之间的关系式（6.13）很容易从上述推论中推导出来。

关于拟阵可表达性问题更深入的讨论，请参阅文献[10.4]。

10.7 二元拟阵

一个拟阵是二元的，如果它在GF(2)上是可表达的，其整数域为模2。

很显然，图G的回路拟阵$M(G)$和割集拟阵$M^*(G)$是二元的。定理4.6已经证明：$M(G)$的每个回路都可表示成G的某些基本回路的环和。对于G的割集，类似的结论也成立。回路和割集的这个性质对于任何二元拟阵都是适用的，然而，它并非普遍适用于任何拟阵。例如，令$S=\{1, 2, 3, 4\}$，并把S的三个元素的子集作为M的回路，那么回路$\{1, 2, 3\}$和$\{1, 2, 4\}$的环和是$\{3, 4\}$，即是M的独立集。

本节，我们建立二元拟阵的几条性质，以便进一步讨论这类拟阵的其它性质。

令M是集合S上的拟阵*，$B=\{b_1, b_2, \cdots, b_r\}$是$M$的基，$B^*=S-B=\{e_1, e_2, \cdots, e_q\}$是$M$的相应的补基。曾记得，$B^*\cap C(e_j)=e_j$，$j=1, 2, \cdots, q$，其中$C(e_j)$是$e_j$的基本回路。类似地，$B\cap C^*(b_j)=b_j$，$j=1, 2, \cdots, r$，其中$C^*(b_j)$是$b_j$的基本回路。在无向图的情况下（第六章），定义$M$对应于基$B$的基本补回路矩阵$D_f$和基本回路矩阵$D_f^*$。显然，这两个矩阵形如

$$\begin{array}{rl} & \begin{array}{cc} b_1b_2\cdots b_r & e_1e_2\cdots e_q \end{array} \\ D_f= & [\quad U_r \quad | \quad F \quad] \\ D_f^*= & [\quad G \quad | \quad U_q \quad] \end{array}$$

注意，$D_f=[d_{ij}]$和$D_f^*=[d_{ij}^*]$都是[0，1]矩阵。

假定矩阵M是二元的，那么它在GF(2)上的正则表示形如

$$\begin{array}{c} \begin{array}{cc} b_1b_2\cdots b_r & e_1e_2\cdots e_q \end{array} \\ [\quad U_r \quad | \quad A \quad] \end{array} \tag{10.10}$$

令基本回路$C(e_j)$是$\{e_j, b_1, b_2, \cdots, b_k\}$。那么式（10.10）中对应于$C(e_j)$的元素的列的模2和是0。因为矢量的模2和相当于对应集合的环和，所以可以看出

$$a_{ij}=\begin{cases} 1, & 若b_i\in C(e_j) \\ 0, & 若b_i\notin C(e_j) \end{cases}$$

也就是

$$A=G^t \tag{10.11}$$

于是，矩阵$[U_r\ G^t]$是M对应于基B在GF(2)上的正则表示。

从对偶拟阵M^*的正则表示出发，我们用相同的方法可以证明：$[F^t\ U_q]$是M^*在GF(2)上的正则表示。

因为一个拟阵对于给定的基有一个唯一的正则表示，由推论10.24.1可知，

$$F=G^t \tag{10.12}$$

于是证明了下面的定理。

定理10.25 令M是集合S上的二元拟阵。

1．M对应于任何基的基本补回路矩阵是M的一种正则表示。

2．M对应于任何基的基本回路矩阵是M^*的一种正则表示。 □

从式（10.12）还可以得到下面的重要结论。

定理10.26 令M是二元拟阵，D_f和D_f^*是M对应于相同基的基本补回路矩阵和基本回路矩阵。那么

$$D_f(D_f^*)^t=0 \tag{10.13}$$ □

给定了拟阵M的回路C，我们就可以把C与一个（0，1）行矢量相关联，这个行矢量的每个元对应于M的一个元素。如果对应的元素在C中，元的值就是1。例如，D_f^*的行是对应于基本回路的矢量。M的所有回路矢量的矩阵称为M的回路矩阵，并记为$D^*(M)$。按照类似的方法，可以定义M的补回路矩阵$D(M)$。

---

* 此段与前面有重复，但为了与下文联系起来，仍保留。——译者注

设x是对应于二元拟阵M的回路C的矢量。因为D_f是M的正则表示，所以D_f对应于C中元素的列的环和是0。即

$$D_f x^t = 0 \tag{10.14}$$

类似地，如果x是补回路矢量，则有

$$D_f^* x^t = 0 \tag{10.15}$$

注意，以上是假定D_f、D_f^*和x的列按相同元素顺序排列。

定理10.27 令M是S上的二元拟阵。对于M的任何基B和回路C，如果$C-B=\{x_1,x_2,\cdots,x_k\}$，并且$C(x_i)$是$x_i$对应于$B$的基本回路，那么

$$C=C(x_1)\oplus C(x_2)\oplus\cdots\oplus C(x_k)$$

证明

证明主要以式（10.12）和（10.16）为依据。参见定理3.7的证明过程。 □

定理10.28 令M是一拟阵，对于M的任意基B和回路C，令$C=C(x_1)\oplus C(x_2)\oplus\cdots\oplus C(x_k)$，这里$\{x_1,x_2,\cdots,x_k\}=C-B$，$C(x_i)$是$x_i$在$B$中的基本回路。那么，$M$的任意两个相异回路的环和$C_1\oplus C_2$包含了$M$的一个回路。

证明

假定$C_1\oplus C_2$不包含回路，并令$C_1\cap C_2=\{x_1,x_2,\cdots,x_p\}$。那么$C_1\oplus C_2=(C_1\cup C_2)-\{x_1,x_2,\cdots,x_p\}$在$M$中是独立的。因而存在基$B\supseteq C_1\oplus C_2$。但是$C_1-B=C_2-B=\{x_1,x_2,\cdots,x_p\}$，所以由定理的假设可知，

$$C_1=C(x_1)\oplus C(x_2)\oplus\cdots\oplus C(x_p)=C_2$$

与$C_1\neq C_2$相矛盾。 □

定理10.29 令M是一拟阵，对于M的任意基B和回路C，令$C=C(x_1)\oplus C(x_2)\oplus\cdots\oplus C(x_k)$，其中$\{x_1,x_2,\cdots,x_k\}=C-B$，$C(x_i)$是$x_i$在$B$中的基本回路。那么$M$是二元的。

证明

令$B=\{b_1,b_2,\cdots,b_r\}$是M的任意基，且令$S-B=\{e_1,e_2,\cdots,e_q\}$。$M$对应于$B$的回路矩阵$D_f^*(M)$形如

$$\begin{array}{cc} b_1b_2\cdots b_r & e_1e_2\cdots e_q \\ \left[\begin{array}{c} A \end{array}\right| & \left.\begin{array}{c} U_q \end{array}\right] \end{array}$$

我们通过证明矩阵

$$\begin{array}{cc} b_1b_2\cdots b_r & e_1e_2\cdots e_q \\ \left[\begin{array}{c} U_r \end{array}\right| & \left.\begin{array}{c} A \end{array}\right] \end{array} \tag{10.16}$$

是M在$GF(2)$上的正则表示来证明定理。

令C是M的回路。如果$C-B=\{e_{i_1},e_{i_2},\cdots,e_{i_k}\}$，那么由定理假设可知，

$$C=C(e_{i_1})\oplus C(e_{i_2})\oplus\cdots\oplus C(e_{i_k})$$

从A的定义可知，式（10.16）中矩阵的列对应C中的元。这个矩阵的列的模2和是零。所以在$GF(2)$上，这些列是线性相关的。

设$W=\{b_1,\cdots,b_t,e_1,\cdots,e_p\}$是由式（10.16）中矩阵所对应的列矢量的线性相关性而

导出的S上的拟阵M'的回路。令

$$W=C(e_1)\oplus C(C_2)\oplus\cdots\oplus C(e_p)$$

由定理10.28，W包含了M的回路C'。但C'不可能是W的真子集，那么这就与W是M'的回路相矛盾。于是$W=C'$。

所以，式（10.16）中的矩阵是M在$GF(2)$上的一种正则表示。 □

下面定理将给出二元拟阵更进一步的特征。

定理10.30 对于拟阵M，下列各种说法是等价的：

1.对于M的任意回路C和补回路C^*，$|C\cap C^*|$是偶数。

2.M的任意相异回路聚合的环和是M的不相交回路的并。

3.对于M的任意基B和回路C，如果$C-B=\{e_1,e_2,\cdots,e_q\}$和$C(e_i)$是$e_i$在基$B$中的基本回路，那么$C=C(e_1)\oplus C(e_2)\oplus\cdots\oplus C(e_q)$。

4.M是二元的。

证明

1⟹2 令$C_1,C_2,\cdots,C_k$为M的不同回路，又令$A=C_1\oplus C_2\oplus\cdots\oplus C_k$，不失一般性，可以假定$A$不含环。

因为对于任意补回路C^* $|C^*\cap C_i|$是偶数，对$1\leqslant i\leqslant k$，所以容易推出$|A\cap C^*|$也是偶数。假定A不是相关的。那么就会产生矛盾，因为根据引理10.3可知，这里存在一个回路，它恰好包含了A中一个元素。故A是相关的，并包含了回路C。

如果$A=C$，定理得证。如果不是，令$A_1=A\oplus C$。注意，对于任何补回路C^*，$|C^*\cap A_1|$是偶数。所以对于A_1可以重复上述的讨论过程。因为A_1是有限的，$A_1=A-C$，所以这个过程最终一定会停止，从而产生了一个边不相交回路的有限聚合，这些回路的并是A。

2⟹3 见定理4.6的证明。

3⟹4 同定理10.29。

4⟹1 由定理10.27，每个回路可用基本回路表示成

$$C=C(x_1)\oplus C(x_2)\oplus\cdots\oplus C(x_k)$$

由式（10.16），对于所有$1\leqslant i\leqslant k$，$|C^*\cap C(x_i)|$是偶数。所以容易推出$|C^*\cap C|$是偶数。 □

显然，在定理10.30中用补回路代替回路，就能得到二元拟阵用补回路表示的特征。

10.8 可定向拟阵

拟阵M是可定向的，如果能够对回路矩阵$D^*=D^*(M)$和补回路矩阵$D=D(M)$中某些非零元素给予负号，并使标号回路矩阵的任一行和标号补回路矩阵的任一行的内积在整数环上是零。

显然，图的回路拟阵和割集拟阵都是可定向的。

定向拟阵M的涂色是把M的元素划分为R、G和B三个集合，并且标明了集合G的一个元素。我们可以把这形象地看作用三种颜色给M的元素涂色，每个元素都可涂成红、绿

或蓝颜色，并且恰好有一个绿元素涂成深绿。注意，深绿元素和绿元素同样看待。

本节的主要结论是Minty在文献[10.2]中提出的“弧着色引理”。

定理10.31（弧着色引理）。设M为可定向拟阵。对于M元素的任何着色，仅有下面的一种论述成立：

1. 存在一个包含深绿元素，但不包含蓝元素的回路，其中所有绿元素是同样定向的（即在标号回路矩阵中具有相同的符号）。

2. 存在包含深绿元素、但不包含红元素的补回路，其中所有绿元素是同样定向的。

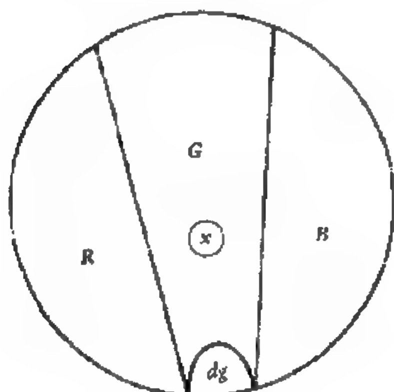


图 10.4

证明

通过对绿元素数目的归纳加以证明。

如果仅有一个绿元素，由公理G-2即可推出结论。

假定当绿元素数目是m时，结论成立。考虑有$m+1$个绿元素的着色。选取一个绿元素但不是深绿元素（见图10.4）。

把元素x染红，在着色结果中有m个绿元素。如果出现有第2种补回路，则定理得证。

假如把x染蓝，如果在形成的着色中有第1种回路，则定理得证。

假如上面两种情况都不发生。那么，由归纳假设，则有

a. 当x染蓝，有第2种补回路。

b. 当x染红，有第1种回路。

现在，把标号回路矩阵和标号补回路矩阵的相应行示出如下：

	dg	R	B	G	X
补回路	$+1$	$0\ 0\ \cdots\ 0\ \ 0$	$1\ -1\ \cdots\ 0\ 1$	$1\ 1\ \cdots\ 1\ 0$	?
回　路	$+1$	$-1\ 1\ \cdots\ 0\ -1$	$0\ \ 0\ \cdots\ 0\ 0$	$0\ 1\ \cdots\ 1\ 0$	?

不失一般性，这里，可以假定$+1$出现在两个矢量的深绿元素位置上。

由定义，这两个矢量的内积是零。深绿元素对内积的贡献是1，红元素和蓝元素的贡献是0；绿元素的贡献是非负整数P，x元素的贡献是未知整数q，其q一定是0、1或-1。于是有$1+P+q=0$，这个方程仅对$P=0$和$q=-1$成立。所以，在两个矢量中，间号处一个是$+1$，另一个是-1。选取间号是1的矢量，就得到所要求的回路或补回路。

于是论断1或论断2成立，但两者不可能同时成立，如果两者同时成立的话，对应的回路和补回路矢量的内积不为零。 □

弧着色引理对图的特殊情况是显然成立的（文献[10.5]，[10.6]），并且在第十一章将用它来证明电阻网络的“无增益”性质。

10.9 拟阵和Greedy算法

考虑集合S，它的元素s_i都有给定的非负权$w(s_i)$。S的子集的权定义为子集中所有元素的权之和。令$\mathscr{I}$为S的子集聚合。组合优化中许多问题可归结为下面的问题：

确定$\mathscr{I}$的元的最大权。

例如，寻找加数连通图G的最大权生成树就可归结为上述的问题，其中$\mathscr{I}$是G的所有生成树的聚合。

下面给出的叫做Greedy算法，就是用来解决这类组合最优化问题。

算法10.1 Greedy算法

S1. 从所有s及$\{s\}\in\mathscr{I}$选取元素s_1，并使$\{s_1\}\in\mathscr{I}$及$w(s_1)\geqslant w(s)$，如果不存在这样的s_1，则停止。

S2. 对于所有$s\neq s_1$及$\{s_1, s\}\in\mathscr{I}$中选取元素s_2，并使$\{s_1, s_2\}\in\mathscr{I}$以及$w(s_2)\geqslant w(s)$，如果不存在这样的$s_2$，则停止。

Sk. 选取与$s_1, s_2, \cdots, s_{k-1}$不相同的元素$s_k$，使$\{s_1, s_2, \cdots, s_k\}\in\mathscr{I}$，且在所有这些$s$中，$w(s_k)$最大。如果不存在这样的$s_k$，则停止。 □

显然，当Greedy算法停止时，就会取出$\mathscr{I}$的一个最大元，但不一定具有最大权。

例如，如果

$$S=\{a, b, c, d\}$$

$$w(a)=4,\ w(b)=3,\ w(c)=2,\ w(d)=2$$

以及

$$\mathscr{I}=\{\{a\}, \{a, c\}, \{b, c, d\}, \{b, d\}\}$$

那么算法将选取$\{a, c\}$。但是，$\{b, c, d\}$是$\mathscr{I}$的最大权元。然而，当我们把权加以修改，并使

$$w(a)=6,\ w(b)=3,\ w(c)=2,\ w(d)=2$$

算法将再次选取$\{a, c\}$，它现在就是$\mathscr{I}$的最大权元。

下面来考察Greedy算法和$\mathscr{I}$的结构之间的关系。

考虑集合S上的拟阵M，$\mathscr{I}$是M的独立集聚合。令

$$I_1=\{a_1, a_2, \cdots, a_m\}$$

和

$$I_2=\{b_1, b_2, \cdots, b_n\}$$

是两个独立集，其中的元素是按权的非递增顺序排列。于是$w(a_1)\geqslant w(a_2)\geqslant\cdots\geqslant w(a_m)$和$w(b_1)\geqslant w(b_2)\geqslant\cdots\geqslant w(b_n)$。如果对于$1\leqslant i\leqslant k-1$，存在某个$k$，并使$w(a_i)=w(b_i)$和$w(a_k)>w(b_k)$，或者对于$1\leqslant i\leqslant n$和$m>n$，有$w(a_i)=w(b_i)$，那么$I_1$就是字典式大于$I_2$。字典式不小于其它任何集合的集称为最大字典式的。由这个定义，显而易见，最大字典式的独立集一定是一个基。

集合$B\in\mathscr{I}$在$\mathscr{I}$中是Gale最优的，如果对于每个$I\in\mathscr{I}$，I和B之间存在一一对应关系，并使得对于所有$a\in I$，有$w(a)\leqslant w(b)$，从而b是B中对应a的元素。显然，只有基才能

是Gale最优的。而且，如果某个基是Gale最优的，那么它一定有最大的权。

以下假定每个集合的元素都是按权的非递增顺序排列。

定理10.32 令$\mathscr{I}$是S上拟阵的独立集聚合，而B是M的基。对于S的元素的任何非负加权，下列论断是等价的：

1. B在$\mathscr{I}$中是最大字典式的。

2. B在$\mathscr{I}$中是Gale最优的。

3. B是$\mathscr{I}$的最大权元。

证明

$1 \Longrightarrow 2$令$B=\{b_1, b_2, \cdots b_r\}$是M的最大字典式基。假定B不是Gale最优的。那么存在独立集$I=\{a_1, a_2, \cdots, a_k\}$，对于$1 \leqslant i \leqslant k-1$，使得$w(a_i)=w(b_i)$和$w(a_k)>w(b_k)$，然后考虑独立集$B_{k-1}=\{b_1, b_2, \cdots, b_{k-1}\}$和$I=\{a_1, a_2, \cdots, a_k\}$。由公理I-3，则有$a_j \in I$，并使得

$$I'=\{b_1, b_2, \cdots, b_{k-1}, a_j\}$$

是独立集。但是，因为$w(a_j) \geqslant w(a_k) > w(b_k)$，所以$I'$是字典式大于$B$，这与$B$在$\mathscr{I}$中是最大字典式相矛盾。

$2 \Longrightarrow 3$ 显然成立。

$3 \Longrightarrow 1$ 令B是最大字典式基，$B=\{b, b_2, \cdots, b_r\}$，$B'=\{b_1', b_2', \cdots b_r'\}$是最大权基，因为$1 \Longrightarrow 2$，则有

$$w(b_i) \geqslant w(b_i'), \text{ 对于所有} 1 \leqslant i \leqslant r$$

因B'是最大数基，故推得$w(b_i)=w(b_i')$。于是B'也是最大字典式的。 □

容易证明（同上述定理中$1 \Longrightarrow 2$的证明）：从Greedy算法取出一个最大字典式基，由上述定理可知，它具有最大权。从而有下面的定理。

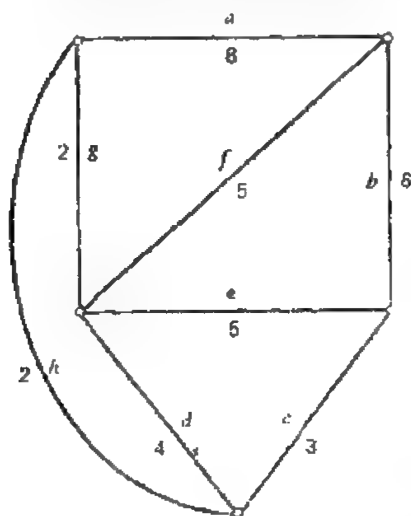


图 10.5 加权图

定理 10.33 令$\mathscr{I}$是S上拟阵的独立集聚合，其中S的元素给予非负权。当把Greedy算法用到$\mathscr{I}$上时，就能够找出$\mathscr{I}$的最大权元。 □

根据上述定理，很明显，如果按权的非递增顺序选取拟阵的元素，就能够找到最大权基，不过当选入某个元素会破坏已选元素集合的独立性时，就把它排除。Greedy算法显然也能找到最小权基。

举例，考虑图10.5的加权图G。边的权已在图中标出。为了求G的最大权生成树，Greedy算法首先把边按权的非递增顺序排列为

$$a, b, e, f, d, c, g, h$$

算法首先选取三条边a, b, e，因为它们不包含任何回路。边f被排除，因为集合$\{a, b, e\} \cup \{f\}$包含回路。接下来取边d。c也被排除，因为它与已经取过的e和d构成了回路。同样原因，g和h也被排除。结果Greedy算法找出$\{a, b, e, d\}$，这是G的最大权生成树。

现在证明定理10.33的逆定理。

定理10.34 令$\mathscr{I}$是集合S的子集聚合，且有$A\in\mathscr{I}$和$B\in A$包含了$B\in\mathscr{I}$的性质。那么，对于S的元素的所有非负加权，只有当$\mathscr{I}$是S上拟阵的独立集聚合，把Greedy算法用在$\mathscr{I}$上才能找出$\mathscr{I}$的最大权元。

证明

由独立性公理，显然只需要证明：如果$A=\{a_1,a_2,\cdots,a_k\}\in\mathscr{I}$和$B=\{b_1,b_2,\cdots,b_{k+1}\}\in\mathscr{I}$，则存在$b_i\notin A$，并使$A\cup b_i\in\mathscr{I}$。

为此，定义S的元素的加权如下：

$$w(a_i)=1,\qquad 1\leqslant i\leqslant k$$

$$w(b_i)=x,\qquad b_i\subset B-A$$

$$w(e)=0,\qquad e\in S-(A\cup B)$$

其中$0<x<1$。那么Greedy算法先取元素$a_1,a_2,\cdots,a_k$。如果不存在使$\{b_i,a_1,\cdots,a_k\}\in\phi$的$b_i$，算法便从$S-(A\cup B)$中取元素。当算法终止时，就取出一个权等于$A$的权的集合。

如果$|B\cap A|=t$，那么

$$w(A)=k$$

以及

$$w(B)=t+(k-1+t)x$$

显然，可以选择$0<x<1$，以使$w(A)<w(B)$。但是，如果这样，算法就不能选出ϕ的最大权元。这与定理相矛盾。 □

寻求最大权生成树问题的Greedy算法，最早是Kruskal发现的。这个算法在拟阵上的推广则是Rado在文献[10.8]、Edmonds在文献[10.9]、Gale在文献[10.10]和Welsh在文献[10.11]中独立发现的。

10.10 进一步阅读

文献[10.4]和[10.12]是两本很出色的教科书，其中包含了丰富的拟阵理论知识。文献[10.13]中有一章讨论拟阵。文献[10.14]对拟阵作了精练的介绍。Wilson通过证明图的边不相交生成树的两个算法，来清楚地说明了拟阵的统一概括功能。除了Whitney[10.1]的精典论文外，我们还极力推荐文献[10.15]直到[10.25]，以及两本书[10.26]和[10.27]。

Tutte在文献[10.3]，[10.7]，[10.18]中创立了链群和拟阵理论。他定义拟阵是正则的，如果它同正则链群的拟阵同构。可以证明，一个拟阵是正则的，当且仅当它可在任何域上表示。另一个性质是，一个拟阵是正则的，当且仅当它是可定向的文献[10.2]。Tutte还导出了拟阵可图示的充要条件。二元拟阵是图示的，当且仅当它不含Fono拟阵，或者不含其对偶、或者不含$M^*(K_5)$、$M^*(K_{3,3})$作为其余因子。这个问题更详尽的讨论，可见文献[10.19]和[10.4]。（关于Fono拟阵的定义，参考习题10.22）

有关拟阵连通性理论，见文献[10.20]。有关适用于一般拟阵的可定向理论，见文献[10.28]和[10.29]。

文献[10.16]利用拟阵概念对香农(Shannon)开关博奕作了解答。可参阅文献[10.30]，[10.31]及[10.4]。

拟阵的美妙之处就在于它的统一性，它已经对横截理论(transversal theory)和图论中的几个结论给出了简明论证。可参见文献[10.24]、[10.30]及[10.14]。

Lawler[10.27]是拟阵算法方面一本很好的教科书。另外还有文献[10.32]，[10.33]及[10.4]。

拟阵理论现在越来越多地用来解决电网络问题。参见文献[10.34]到[10.43]。[10.37]是一篇很好的拟阵理论导引文献。

10.11 习　题

10.1 令M是S上的拟阵，$A\subseteq S$。定义$\mathscr{I}'$是S的子集X的聚合，并使得X在M中是独立的，$X\cap A=\varnothing$。证明：$\mathscr{I}'$是S上拟阵的独立集聚合。

10.2 令S是具有n个元素的集合。证明：S中所有含有k个元素或者小于k个元素的子集聚合$\mathscr{I}$是拟阵的独立集集合。这个拟阵叫作秩为k的均匀拟阵，记为$U_{k,n}$。

10.3 令M是S上的拟阵，B_1、B_2是M的两个不同的基。证明：B_1和B_2之间存在着一一对应关系，并使得对于所有$e\in B_1$，$(B_2-e')\cup e$是M的基，这里e'是B_2中对应e的元素。

10.4 如果B_1、B_2是拟阵M的基，且$X_1\subseteq B_1$，试证存在$X_2\subseteq B_2$，并使得$(B_1-X)\cup X_2$和$(B_2-X_2)\cup X_1$都是M的基(文献[10.44])。(也可见[10.12]，第五章)

10.5 令$\mathscr{C}$是S的非零子集的聚合，且对于$\mathscr{C}$中任何两个不同元X、Y，$x\in X\cup Y$，$y\in X-Y$，则存在$Z\in\mathscr{C}$，并使得$y\in Z(X\cup Y)-x$。那么求证：$\mathscr{C}$中最小元的集合$\mathscr{C}'$是拟阵的回路集合(文献[10.13])。

10.6 如果C是拟阵M的回路，$a\in C$，则证明：存在基B，并使得$C=C(a, B)$。

10.7 如果B是拟阵M的基，且$x\in B$，则证明：恰好有一个补回路C^*，并使得$C^*\cap(B-x)=\varnothing$。

10.8 如果C是拟阵M的回路，x、y是C中不同的元素，则证明：存在包含x和y，但不包含C的其它元素的补回路(文献[10.2])。

10.9 令M是S上的拟阵，x、y、z是S中不同的元素。如果存在包含x和y的回路C和包含y和z的回路C_1，那么证明：存在包含x和z的回路C_2。

10.10 集合$A\subseteq S$在S上的拟阵M中是闭合的，如果对所有$x\subset S-A$，$\rho(A\cup x)=\rho(A)+1$。试证：两个闭集的交仍是闭集。

10.11 令M是集合S上的拟阵。S的子集A的闭包$\sigma(A)$是S中所有具有性质$\rho(A\cup x)=\rho(A)$的元素x的集合。证明：

(a) 如果x包含在$\sigma(A\cup y)$中，但不在$\sigma(A)$中，那么y必在$\sigma(A\cup x)$中。

(b) 元素x属于A，当且仅当$x\subset A$，或者M中存在回路C，而$C-A=\{x\}$。

10.12 S上的拟阵M的超平面是S的最大真闭子集。证明：H是M的超平面，当且仅当$S-H$是M的补回路。(见文献[10.4]，用超平面定义的公理系统)。

10.13 证明：如果M是S上的拟阵，且$A\subseteq S$，那么M到T的收缩$M\cdot T$就是拟阵，它的补回路恰好是M包含在A中的那些补回路。

10.14 如果M是S上的拟阵和$T\subseteq X\subseteq S$，证明：

(a) $M|T=(M|X)|T$。

(b) $M\cdot T=(M\cdot X)\cdot T$。

(c) $(M|X)\cdot T=(M\cdot(S-(X-T)))|T$。

(d) $(M\cdot X)T=(M|S.(X-T))\cdot T$。

10.15 如果对于S的每一对不同元素x和y，M中都存在包含x和y的回路，那么S上的拟阵M就是连通的，或者是不可分的。否则，它就是不连通的，或者是可分的。证明：M是连通的，当且仅当它的对偶M^*是连通的。

注：如果G是图，那么$M(G)$是连通的，当且仅当G是2-连通的。

10.16 证明：S上的拟阵M是不连通的，当且仅当存在S的真子集A，并使得

$$\rho(A)+\rho(S-A)=\rho(S)$$

（见文献[10.1]）。

10.17 说明下述说法正确与否：图G可以收缩到H，当且仅当$M(G)$以$M(H)$为收缩余因子。

10.18 证明：均匀拟阵$U_{2,4}$可在除$GF(2)$外的每一个域上表示出来。

10.19 证明：拟阵是二元的，当且仅当对任何回路C和补回路C^*，$|C\cap C^*|\neq 3$。

10.20 令$\mathscr{F}$是集合S的有限非空子集族。$\mathscr{F}$的子族的横截称为$\mathscr{F}$的部分横截。证明：如果$\mathscr{F}$是集合S的有限非空子集族合，那么$\mathscr{F}$的部分横截的集合是S上的拟阵的独立集集合（见8.6节横截的定义）。求出这个拟阵的秩函数。如果存在S的子集族$\mathscr{F}$，并使得$\mathscr{I}(M)$是$\mathscr{F}$的部分横截族，则S上的拟阵M称为横截拟阵。求出横截拟阵的秩函数（见定理8.15）。

10.21 证明：每个k-均匀拟阵是横截拟阵。

10.22 Fano拟阵F是定义在集合$S=\{1,2,3,4,5,6,7\}$上的拟阵，其中F的基是除了$\{1,2,3\}$、$\{1,4,5\}$、$\{1,6,7\}$、$\{2,4,7\}$、$\{2,5,6\}$、$\{3,4,6\}$和$\{3,5,7\}$以外S的所有含有三个元素的子集。证明：F是

(a) 二元的，

(b) 非横截的，

(c) 非图的，

(d) 非补图的。

10.23 证明：K_4的回路拟阵不是横截拟阵。

10.24 如果S可以表示为不相交回路的并，则S上的拟阵是欧拉的。如果M的每一个回路都有偶数个元素，则拟阵是二分的。证明：拟阵是二分的，当且仅当M^*是欧拉的。

10.25 令D是无自环的有向图，X和Y是D的顶点的两个不相交集合。如果存在A条从A到Y的顶点不相交链，则X的子集A是独立的。证明：这些独立集构成了X上拟阵的独立集（这样的拟阵称为Gammoid。）（见文献[10.46]）。

10.26 S上的拟阵M是基可排列的，如果对M的两个基B_1和B_2，B_1和B_2之间存在一一对应关系，从而对于每个$x\in B_1$，$(B_1-x)\cup x'$和$(B_2-x')\cup x$都是M的基。x'是B_2中对应于x的元素。证明：

(a) $M(K_4)$不是基可排列的。

(b) 如果M在S和$T\subseteq S$上是基可排列的，那么$M|T$也是基可排列的。

(c) 如果M是基可排列的，那么M的每一个余因子也是基可排列的。

10.27 令M_1和M_2是S上的两个拟阵。

(a) 证明M_1和M_2的独立集I和J的所有并$I\cup J$的集合构成了一个新拟阵的独立集（这个拟阵称为M_1和M_2的并，记为$M_1\cup M_2$）。

(b) 如果ρ_1和ρ_2代表S上的拟阵M_1和M_2的秩函数，则证明：

$$\rho(A)=\min_{X\subseteq A}\{\rho_1(X)+\rho_2(X)+|A-X|\}$$

其中$A\subseteq S$，ρ是$M_1\cup M_2$的秩函数（文献[10.14]，pp.154—158）。

10.28 令M是S上的拟阵。证明：

（a） M有k个不相交的基，当且仅当对于任何$A\subseteq S$，

$$\rho(A)+|S-A|\geqslant k\rho(S)$$

（b） S可用于不多于k个独立集的并表示，当且仅当对于任何$A\subseteq S$，

$$k\rho(A)\leqslant|A|$$

提示：考虑M的k个部分的并，并利用习题10.27的结论（文献[10.14]，pp.154—158）。

10.29 证明：当Greedy算法选取了k个元素时，在所有有k个或更少元素的独立集中，这k个元素具有最大数。

10.30 一组工作可以在一台机器上做。所有工作需要花费相同的工时。每项工作都给定了截止期限。

（a） 证明：那些能够按时完成工作的子集聚合构成了拟阵的独立集。

（b） 假如每项工作到期不能完成，就要被罚款。试求按什么次序做这些工作才能使总的罚款最少？

10.31 令M是拟阵，它的每个元素都给定一个非负数。证明：

（a） 最大数基中没有一个元素是M的任一回路中的最小元素。

（b） 最大数基中每一个元素至少是M的一个补回路中的最大元素。用（b）来设计一个构造拟阵的最大数基的程序。（文献[10.46]曾叙述过构造连通图的最小数生成树的程序。）

10.32 令M是S上的拟阵，S中的元素都给定一个非负数。令$\mathscr{B}$是M的基的集合，$\mathscr{C}^*$是M的补回路的集合。证明：

$$\min_{B\in\mathscr{B}}\max_{e\in B}w(e)=\max_{C^*\in\mathscr{C}^*}\min_{e\in C^*}w(e)$$

10.12 参考文献

10.1 H. Whitney, "On the Abstract Properties of Linear Dependence," *Am. J. Math.*, Vol. 57, 509—533(1935).

10.2 G. J. Minty, "On the Axiomatic Foundations of the Theories of Directed Linear Graphs, Electrical Networks and Network Programming," *J. Math. and Mech.*, Vol. 15, 485—520(1966).

10.3 W. T. Tutte, *Introduction to the Theory of Matroids*, American Elsevier, New York, 1971.

10.4 D. J. A. Welsh, *Matroid Theory*, Academic Press, New York, 1976.

10.5 G. J. Minty, "Monotone Networks," *Proc. Roy. Soc.*, *A*, Vol. 257, 194—212 (1960).

10.6 G. J. Minty, "Solving Steady-State Non-Linear Networks of 'Monotone' Elements," *IRE Trans. Circuit Theory*, Vol. CT-8, 99—104(1961).

10.7 J. B. Kruskal, "On the Shortest Spanning Subgraph of a Graph and the Travelling Salesman Problem," *Proc. Am. Math. Soc.*, Vol. 7, 48—49(1956)

10.8 R. Rado, "Note on Independence Functions," *Proc. London Math. Soc.*, Vol. 7, 300—320(1957)

10.9 J. Edmonds, "Matroids and the Greedy Algorithm," *Math. Programming*, Vol. 1, 127—136(1971)

10.10 D. Gale, "Optimal Assignments in an Ordered Set: An Application of Matroid

Theory," *J. Combinatorial Theory*, Vol. 4, 176—180(1968).

10.11 D. J. A. Welsh, Kruskal's Theorem for Matroids," *Proc. Cambridge Phil. Soc.*, Vol. 64, 3- 4(1968).

10.12 Rabe von Randow, *Introduction to the Theory of Matroids*, Springer Lecture Notes in Mathematical Economics, Vol. 109, 1975.

10.13 C. Berge, *Graphs and Hypergraphs*, North Holland, Amsterdam, 1973

10.14 R. J. Wilson, *Introduction to Graph Theory*, Oliver and Boyd, Edinburgh, 1972.

10.15 R. Rado, "A Theorem on Independence Relations," *Quart, J. Math* (Oxford), Vol. 13, 83—89(1942).

10.16 A. Lehman, "A Solution of the Shannon Switching Game," *SIAM J. Appl Math.*, Vol. 12, 687—725(1964).

10.17 W. T.Tutte, "A Homotopy Theorem for Matroids-I and II," *Trans. Am. Math Soc.*, Vol. 88, 144—174(1958).

10.18 W. T. Tutte, "Lectures on Matroids," *J. Res. Nat. Bur. Stand*, Vol 69B, 1-48 (1965).

10.19 W. T. Tutte, "Matroids and Graphs," *Trans. Am. Math. Soc.*, Vol. 90, 527—552 (1959).

10.20 W. T. Tutte, "Connectivity in Matroids," *Canad. J. Math.*, Vol. 18, 1301—1324 (1966).

10.21 L. Mirsky, "Application of the Notion of Independence to Combinatorial Analysis," *J. Combinatorialr Theory*, Vol. 2, 327—357(1967).

10.22 F. Harary and D. J. A. Welsh, "Matroids versus Graphs," in *The Many Facets of Graph Theory*, Springer Lecture Notes, Vol. 110, 1969, pp. 155—170

10.23 R. J. Wilson, "An Introduction to Matroid Theory," *Am Math Monthly*, Vol. 80, 500—525(1973).

10.24 J. Edmonds, "Minimum Partition of a Matroid into Independent Subsets," *J. Res. Nat. Bur. Stand.*, Vol. 69B, 67—72(1965).

10.25 J. Edmonds and D. R. Fulkerson, "Transversals and Matroids Partition," *J. Res. Nat. Bur. Stand.*, Vol. 69B, 147—153(1965).

10.26 L. Miraky, *Transversal Theory*, Academic Press, London, 1971.

10.27 E. L. Lawler, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York, 1976.

10.28 R. G. Bland and M.Las Vergnas, "Orientability of Matroids," *J Combinatorial Theory B*, Vol. 24, 94—123(1978).

10.29 J. Folkman and J. Lawrence, "Oriented Matroids," *J Combinatorial Theory B*, Vol. 25, 199—236(1978).

10.30 J. Edmonds, "Lehman's Switching Game and a Theorem of Tutte and Nash-Williams," *J. Res. Nat. Bur Stand*, Vol 69B, 73—77(1965)

10.31 J Bruno and L. Weinberg, "A Constructive Graph-Theoretic Solution of the Shannon Switching Game," *IEEE Trans Circuit Theory*, Vol CT-17, 74—81

10.32 D.E. Knuth,"Matroid Partitioning,"Stanford University Rep STAN-CS-73-342, 1—12(1973).

10.33 J. Edmonds,"Matroid Partition," in *Lectures in Appl. Math* Vol. 11: *Mathematics of Decision Scienes*, 1967, pp. 335—346

10.34 R.J Duffin," Topology of Series-Parallel Networks," *J. Math. Anal., Appl* Vol.10, 303—318(1965).

10.35 R.J. Duffin and T. D. Morley,"Wang Algebra and Matroids," *IEEE Trans. Circuits and Syst.*, Vol. CAS-25, 755—762(1978).

10.36 H Narayanan,"Theory of Matroids and Network Analysis," Ph. D. Thesis, Indian Instiute of Technology, Bombay, India, 1974

10.37 J Bruno and L. Weinberg,"Generalized Networks: Networks Embedded on a Matroid, Part 1," *Networks*, Vol. 6, 53—94(1976).

10.38 J Bruno and L Weinberg,"Generalized Networks: Networks Embedded on a Matroid, Part 2." *Networks*, Vol 6, 231—272(1976).

10.39 L Weinberg, "Matroids, Generalized Networks and Electric Network Synthesis," *J Combinatorial Theory B*, Vol. 23, 106—126(1977).

10.40 M. Iri and N. Tomizawa,"A Unifying Approach to Fundamental Problems in Network Theory by Means of Matroids," *Electron. Commun in Japan*, Vol.58-A, 28 —35(1975)

10.41 A Recski, "On Partitonal Matroids with Applications," in *Coll. Math Soc J. Bolyai*, Vol. 10: *Infinite and Finite Sets*, North-Holland-American Elsevier, Amsterdam, 1974, pp. 1169—1179

10.42 A Recski, "Matroids and Independent State Variables," *Proc. 2nd European Conf. Circuit Theory and Design*, Genova, 1976.

10.43 E Petersen, "Investigating Solvability and Complexity of Linear Active Networks by Means of Matroids," *IEEE Trans. Circuits and Syst.*, Vol CAS-26, 330—342(1979)

10.44 C.Greene,"A Multiple Exchange Property for Bases," *Proc Am Math Soc.*, Vol 39, 45—50(1973).

10.45 J. H. Mason,"On a Class of Matroids Arising from Paths in Graphs," *Proc London Math. Soc.*, Vol.25,55- 74(1972)

10.46 R.C.Prim,"Shortest Connection Networks and Some Generalizations," *Bell Sys. Tech. J.*, Vol. 36, 1389—1402(1957).

第二篇　电 网 络 理 论

第十一章　图 和 网 络

电网络就是网络元件如电阻、电容、电感、电压源和电流源的互相连接。每个网络元件都与两个变量有关，即电压变量$v(t)$和电流变量$i(t)$。因为它们都是时间的函数，而且在时间过程中可取正、负值，所以我们就需要确定这些变量的参考方向，给每个网络元件画上箭头，称为定向，这样就确定了参考方向（图11·1）。这个箭头意味着当电流与箭头同向时，电流$i(t)$为正。进一步假设电压$v(t)$的正极为箭头的尾端，当网络元件的电压降与箭头同向时，则电压$v(t)$为正。



图 11.1　网络元件(表示法)

网络元件的特性可用元件上的电压和电流变量之间的物理关系确定。注意有些网络元件上的电压变量可取确定值，而另一些电流变量可取确定值。这样的元件分别称为电压源和电流源。

网络理论中的两个基本定律是基尔霍夫定律，可表述如下：

基尔霍夫电流定律（KCL）：流出某节点的电流代数和为零。

基尔霍夫电压定律（KVL）：环绕某回路的电压代数和为零。

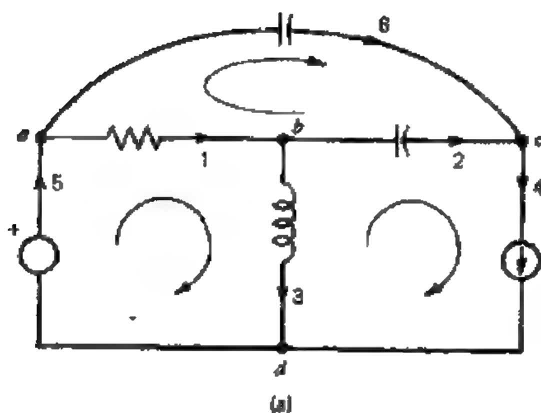
例如，图11.2(a)所示网络的KCL、KVL方程给定如下：

KCL方程

节点a　$i_1-i_5+i_6=0$

节点c　$-i_2+i_4-i_6=0$

节点b　$-i_1+i_2+i_3=0$



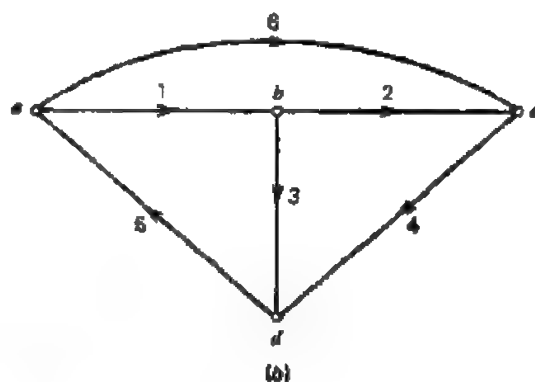


图 11.2　网络有向图表示法
（a）网络N；（b）N的有向图。

KVL方程

回路{1, 3, 5}　　$v_1+v_3+v_5=0$

回路{2, 4, 3}　　$v_2+v_4-v_3=0$

回路{1, 6, 2}　　$-v_1+v_6-v_2=0$

给定一个电网络N，网络分析的问题就是确定符合基尔霍夫定律和符合表征该网络各元件特性的电压-电流关系的各元件电压和电流。

注意：运用基尔霍夫定律列出的方程，本质上是代数式，它们仅取决于网络元件互相连接的方式，而与网络元件的本质无关。电网络的有些性质仅取决于网络的结构。在研究这些性质时，可以很方便地将网络元件看成与两个变量$v(t)$，$i(t)$有关的有向边。因此，我们可以认为电网路是个有向图，它的每个边与两个变量$v(t)$和$i(t)$有关，而且必须满足基尔霍夫定律和某种特定的物理关系。

例如，与图11.2(a)所示网络对应的有向图，如图11.2(b)所示。

现在很容易地看出，对于一个网络N，其KCL和KVL方程可分别写成

$$Q_c I_e=0$$

和

$$B_c V_e=0$$

这里Q_c和B_c是与N对应的有向图的割集和回路矩阵，I_e和V_e分别是N的电流和电压列矢量。

在本章以及下面两章的讨论中，我们在表示电网络和对应的有向图时，使用了相同的符号。在大多数情况下，一个图可认为是一个网络，反之亦然。因此，所有关于图的定义（例如连通性，零度和秩）也可用在网络中。我们也可把顶点称做节点。

本章，我们将讨论以图论为主要依据的网络分析的若干问题。其主要目的在于阐明图论在系统地建立网络方程和发现某些网络的基本性质中的应用。

11.1 回路和割集的变换

本节我们将研究一个电网络N的元件电流间以及元件电压间的关系，这些关系呈现出基尔霍夫两个定律以及N的回路和割集矩阵之间的正交关系。不失一般性，我们可假设N是连通的。

设T是N的一个生成树，B_f和Q_f是相对于T的N的基本回路矩阵和基本割集矩阵。那么基尔霍夫电流和电压方程变为

$$Q_f I_e=0 \qquad (11.1)$$

和

$$B_f V_e=0 \qquad (11.2)$$

假设我们将I_e和V_e划分为

$$I_e=\begin{bmatrix} I_c \\ I_t \end{bmatrix}$$

和

$$V_e=\begin{bmatrix} V_c \\ V_t \end{bmatrix}$$

这里对应于T的连支和树支矢量分别用下标c和t来区分。那么式（11.1）和（11.2）可写成

$$[Q_{fc}\ U]\begin{bmatrix} I_c \\ I_t \end{bmatrix}=0 \tag{11.3}$$

和

$$[U\ B_{ft}]\begin{bmatrix} V_c \\ V_t \end{bmatrix}=0 \tag{11.4}$$

回忆式（6.13）

$$Q_{fc}=-B_{ft}^{t} \tag{11.5}$$

首先考查式（11.3）。从这个方程得到

$$\begin{aligned} I_t &= -Q_{fc}I_c \\ &= B_{ft}^{t}I_c \end{aligned} \tag{11.6}$$

所以可把I_e表示为：

$$\begin{aligned} I_e &= \begin{bmatrix} I_c \\ I_t \end{bmatrix} \\ &= \begin{bmatrix} U \\ B_{ft}^{t} \end{bmatrix} I_c \\ &= B_f^{t} I_c \end{aligned} \tag{11.7}$$

从式（11.4）起，我们用相同的方法可得到

$$V_e=Q_f^{t}V_t \tag{11.8}$$

因此，我们得出如下定理。

定理11.1

1.电网络N中的所有元件电流可表示为连支电流的线性组合，即N的生成树的连支电流的线性组合。

2.电网络N中的所有元件电压可表示为树支电压的线性组合，即N的生成树的树支电压的线性组合。 □

为了说明式（11.7）和（11.8），考虑图11.2所示的网络N。

生成树T包含元件1、4和5，其对应T的B_f和Q_f为

$$B_f=\left[\begin{array}{ccc|ccc} \overset{2}{1} & \overset{3}{0} & \overset{6}{0} & \overset{1}{1} & \overset{4}{1} & \overset{5}{1} \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right]$$

$$
Q_f = \begin{array}{c} \begin{matrix} \;2 & \;\;3 & \;\;6 & \;1 & \;4 & \;5 \end{matrix} \\ \left[\begin{array}{ccc|ccc} -1 & -1 & 0 & 1 & 0 & 0 \\ -1 & 0 & -1 & 0 & 1 & 0 \\ -1 & -1 & -1 & 0 & 0 & 1 \end{array}\right] \end{array}
$$

因而，I_e和V_e可表示为

$$
\begin{bmatrix} i_2 \\ i_3 \\ i_6 \\ i_1 \\ i_4 \\ i_5 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} i_2 \\ i_3 \\ i_6 \end{bmatrix}
$$

$$
\begin{bmatrix} v_2 \\ v_3 \\ v_6 \\ v_1 \\ v_4 \\ v_5 \end{bmatrix} = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 0 & -1 \\ 0 & -1 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_4 \\ v_5 \end{bmatrix}
$$

定理11.2 令N是秩为ρ、零度为μ的电网络。设B是由N的回路矩阵中任意μ个独立行组成的矩阵，Q是由N的割集矩阵中任意ρ个独立行组成的矩阵。

1. 回路变换 列矢量I_e满足N的KCL方程，当且仅当存在μ个元素的列矢量I'，并使得

$$I_e = B^t I' \tag{11.9}$$

2. 割集变换 列矢量V_e满足N的KVL方程，当且仅当存在ρ个元素的列矢量V'，并使得

$$V_e = Q^t V' \tag{11.10}$$

证明

1. 设B_f为N的基本回路矩阵，那么就存在一个非奇异矩阵D，并使得

$$B_f = DB \tag{11.11}$$

如果I_e满足KCL方程，那么由式（11.7）有

$$I_e = B_f^t I_e$$

将式（11.11）代入上式，并令$I' = D^t I_e$，则得

$$I_e = B^t I'$$

相反地，如果I'存在，并使得

$$I_e = B^t I'$$

那么

$$QI_e = (QB^t)I'$$

根据定理6.6， $$QI_e = 0$$

则KCL是满足的。

2.可以用对偶方式来证明。 □

方程式（11.9）和（11.10）分别称为回路变换和割集变换。I'和V'的元素分别称为回路变量和割集变量。通常，回路和割集变量分别是连支电流和树支电压的线性组合。然而，如果在回路（割集）变换中使用$B_f(Q_f)$，那么连支电流（树支电压）就成为回路（割集）变量。

下列变换

$$V_e = A^t V'$$

称为节点变换，这里A是关联矩阵。很显然，这是割集变换的特例。如果v_r是A的参考节点，那么V'的各项就可认为是所有节点(v_r除外)对U_r的电压。这些电压通常称为节点-参考点电压，简称网络节点电压。

我们用Tellegen文献[11.1]提出的一个有趣结论来结束本节，这个结论很容易由回路和割集变换推导出来。

定理11.3(TELLEGEN) 考虑二个电网络N和$\hat{N}$，两个网络所对应的图相同。设V_e和Ψ_e分别表示N和$\hat{N}$的元件电压矢量，设I_e和Λ_e为相应的元件电流矢量。那么

1.$V_e^t \Lambda_e = 0$

2.$I_e^t \Psi_e = 0$

证明

1.设B_f和Q_f表示N对应于生成树T的基本回路和基本割集矩阵。因为N的图与$\hat{N}$的图相同，很显然，B_f和Q_f也是$\hat{N}$对应于T的矩阵。

根据回路和割集变换，我们有

$$V_e = Q_f^t V_t$$

和

$$\Lambda_e = B_f^t \Lambda_c$$

所以

$$V_e^t \Lambda_e = V_t^t (Q_f B_f^t) \Lambda_c$$

根据定理6.6

$$V_e^t \Lambda_e = 0$$

2.可用对偶法来证明。 □

特勒根（Tellegen）定理是网络理论的一个十分奥妙的结论，而且得到了应用，请见文献[11.2]、[11.3]。在第十三章，我们将讨论特勒根定理在用伴随网络概念计算网络灵敏度中的应用。请参看文献[11.4]和[11.5]，也可看文献[11.6]。

11.2 回路和割集系统方程组

如同我们以前提到的那样，网络分析的问题就是确定一个电网络中有关元件上的电压和电流。这些电压和电流由基尔霍夫方程和元件的电压-电流（简称v-i）关系确定。然而，这些方程包含有大量的变量。正象我们在定理11.1中看到的那样，并非所有的变量都

是独立的。进而，在定理11.3中清楚可见，我们能够用仅包括连支电流作为变量的回路变换来替代KCL方程。类似地，我们能够用仅包括支路电压作为变量的割集变换来取代KVL方程。我们将利用这些变换的优点，使不同的网络系统方程组中仅包括电压和/或电流的子集作为变量。这样的两种方程组，称为回路和割集系统方程，就是在本节中引出的。为了得到这样的回路系统方程组，我们用回路变换代替KCL，而且在这种情况下，回路变量将作为独立变量。为了得到这样的割集系统，我们用割集变换代替KVL，而且在这种情况下，割集变量将作为独立变量。

考虑一个连通的电网络N。我们假设N仅由电阻(R)、电容(C)和电感(L)包括互感、独立电压源和电流源所组成。并假设电感中的初始电流和电容上的初始电压用适当的电源代替。进而，所有电压和电流变量都是复频变量的拉普拉斯变换。

在N中，不可有仅包含独立电压源的回路。因为如果这样含源的回路存在，那么，根据KVL，将存在相应电压间的线性关系，从而破坏了这个电压源的独立性。出于同样原因，在N中，不能有仅包含独立电流源的割集。这样，根据定理11.12，在N中存在一个包含所有电压源、但不包含电流源的生成树。这样的树是我们推出回路和割集方程系统的起点。

我们首先推导回路系统。

设T为给定网络的一个生成树，而且T包含了所有的电压源但不包含电流源。让我们划分元件电压矢量V_e和元件电流矢量I_e如下：

$$V_e=\begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix}$$

和

$$I_e=\begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix}$$

这里下标1、2和3分别表示对应于电流源、RLC元件、电压源的矢量。设B_f是N对应于树T的基本回路矩阵。对应于N的KVL可写成

$$B_fV_e=\begin{bmatrix} U & B_{12} & B_{13} \\ 0 & B_{22} & B_{23} \end{bmatrix}\begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix}=[0]$$

即

$$V_1=-B_{12}V_2-B_{13}V_3 \tag{11.13}$$

和

$$B_{22}V_2=-B_{23}V_3 \tag{11.14}$$

我们可用回路变换来替代KCL：

$$\begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} U & 0 \\ B_{12}^t & B_{22}^t \\ B_{13}^t & B_{23}^t \end{bmatrix} \begin{bmatrix} I_1 \\ I_l \end{bmatrix}$$

这里I_l表示T的无源连支的电流矢量。我们从这个方程得到

$$I_2 = B_{12}^t I_1 + B_{22}^t I_l \qquad (11.15)$$

$$I_3 = B_{13}^t I_1 + B_{23}^t I_l \qquad (11.16)$$

注意：在连支电流中，仅在I_l中的那些电流是需要确定的。

如果Z_2是RLC元件的阻抗矩阵，那么这些元件的v-i关系可写成：

$$V_2 = Z_2 I_2 \qquad (11.17)$$

在式（11.14）中代入式（11.17），则得

$$B_{22} Z_2 I_2 = -B_{23} V_3$$

在上式中代入式（11.15）并整理，则得

$$(B_{22} Z_2 B_{22}^t) I_l = -B_{23} V_3 - B_{22} Z_2 B_{12}^t I_1 \qquad (11.18)$$

上式就是仅包含$\mu - n_c$个变量的回路系统方程组，这里n_c是N中电流源的数目。注意：$\mu - n_c$等于移去N中所有电流源后所得网络的零度。

式（11.18）中的矩阵

$$Z_l = B_{22} Z_2 B_{22}^t$$

称为N的回路-阻抗矩阵。如果网络N没有互感，那么Z_2将是对角线上没有0项的对角矩阵。因而，在这种情况下，因为B_{22}具有的最大秩等于$\mu - n_c$，所以Z_l将是非奇异的。如果N中有互感，则仅当Z_2正实的，Z_l才是非奇异的。

一旦我们用式（11.18）确定了I_l，那么我们就能够用式(11.15)来确定I_2，并且用式（11.17）来确定V_2。然后用式（11.13）和（11.16）来确定V_1和I_3（注意，I_1和V_3具有给定的值）。因此利用回路系统方程组就可完成对网络N的分析。

注意：回路分析法是先将元件的v-i关系代入KVL方程，然后用回路变换而导出的。用完全对偶方法，我们可推导割集系统方程组如下。

我们先写出KCL方程的分块形式：

$$\begin{bmatrix} Q_{11} & Q_{12} & 0 \\ Q_{21} & Q_{22} & U \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = [0]$$

这里的系数矩阵与N对应于T的基本割集矩阵是一样的。

根据这些方程，则得

$$Q_{12} I_2 = -Q_{11} I_1, \qquad (11.19)$$

$$I_3 = -Q_{21} I_1 - Q_{22} I_2 \qquad (11.20)$$

用割集变换替代KVL方程：

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} Q_{11}^t & Q_{21}^t \\ Q_{12}^t & Q_{22}^t \\ 0 & U \end{bmatrix} \begin{bmatrix} V_b \\ V_3 \end{bmatrix}$$

这里V_b是T的无源支路的电压矢量。根据这些方程，则得

$$V_1=Q_{11}^tV_b+Q_{21}^tV_3 \tag{11.21}$$

$$V_2=Q_{12}^tV_b+Q_{22}^tV_3 \tag{11.22}$$

如果Y_2是N的RLC元件的导纳矩阵，那么元件的v-i关系就能写成

$$I_2=Y_2V_2 \tag{11.23}$$

现在推导V_b与I_1、V_3关系的割集方程组，就做为一个直接了当的练习了。

在式（11.19）中代入式（11.23），则得

$$Q_{12}Y_2V_2=-Q_{11}I_1$$

在上式中代入式（11.22），则得

$$(Q_{12}Y_2Q_{12}^t)V_b=-Q_{11}I_1-Q_{12}Y_2Q_{22}^tV_3 \tag{11.24}$$

上式就是包含了$\rho-n_v$个变量的割集系统方程组，这里n_v是N中电压源的数目。注意：$\rho-n_v$等于从N中去掉所有电压源后所得网络的秩。

式（11.24）中的矩阵

$$Y_b=Q_{12}Y_2Q_{12}^t$$

称为N的割集-导纳矩阵。如果N中无互感，这个矩阵将是非奇异的。如果N中有互感，则仅当Y_2是正实的，Y_b才是非奇异的。

一旦我们用式（11.24）确定了V_b，那么我们就能够用式（11.22）来确定V_2，然后用式（11.23）来确定I_2，最后根据式（11.20）和（11.21）来分别确定I_3和V_1。

现举例说明回路和割集系统方程组的建立过程。

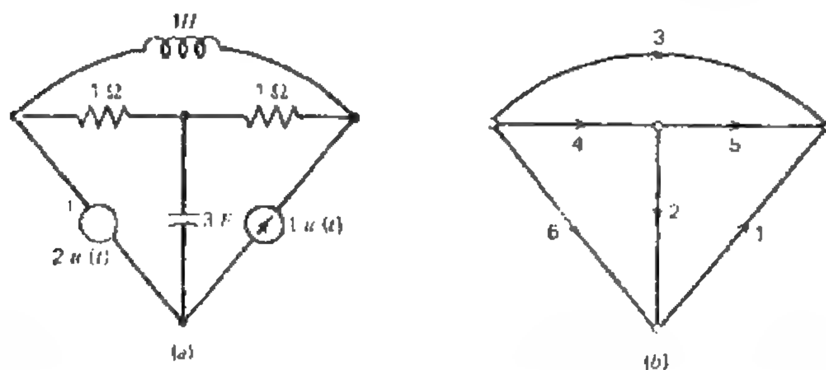


图 11.3 网络和图

考虑如图11.3(a)所示的网络，这里$u(t)$是单位阶跃函数。这个网络的图，如图11.3(b)所示。我们选择生成树T是由边4、5和6组成的。注意：T包含电压源而不包含电流源。对于T的基本回路和基本割集矩阵，按所需分块形式给出如下：

$$B_f=\begin{array}{c|cccc|c} 1 & 2 & 3 & 4 & 5 & 6 \\ \hline 1 & 0 & 0 & -1 & -1 & 1 \\ \hline 0 & 1 & 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 & -1 & 0 \end{array}$$

$$
Q_f=\left[\begin{array}{c|cccc|c} 1 & -1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ \hline -1 & 1 & 0 & 0 & 0 & 1 \end{array}\right]
$$

从这些矩阵中得到

$$B_{12}=[\,0\quad 0\quad -1\quad -1\,]$$

$$B_{13}=[\,1\,]$$

$$B_{22}=\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & -1 & -1 \end{bmatrix}$$

$$B_{23}=\begin{bmatrix} -1 \\ 0 \end{bmatrix}$$

$$Q_{11}=\begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$Q_{12}=\begin{bmatrix} -1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

$$Q_{21}=[\,-1\,]$$

$$Q_{22}=[\,1\quad 0\quad 0\quad 0\,]$$

此外还有

$$Z_2=\begin{bmatrix} 1/3s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Y_3=\begin{bmatrix} 3s & 0 & 0 & 0 \\ 0 & 1/s & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

和

$$v_6(s)=\frac{2}{s}$$

$$i_1(s)=\frac{1}{s}$$

在式（11.28）和（11.24）中代入上式，所得回路和割集系统方程组如下：

回路系统

$$\begin{bmatrix} 1+\dfrac{1}{3s} & -1 \\ -1 & s+2 \end{bmatrix}\begin{bmatrix} i_2(s) \\ i_3(s) \end{bmatrix}=\begin{bmatrix} -\dfrac{3}{s} \\ \dfrac{2}{s} \end{bmatrix}$$

割集系统

$$\begin{bmatrix} 3s+1+\frac{1}{s} & \frac{1}{s} \\ \frac{1}{s} & 1+\frac{1}{s} \end{bmatrix} \begin{bmatrix} v_4(s) \\ v_5(s) \end{bmatrix} = \begin{bmatrix} 6-\frac{1}{s} \\ -\frac{1}{s} \end{bmatrix}$$

假设一个网络N没有独立电压源，那么用节点电压做为独立变量就能方便地描述N，其方法如下：

设A是N的关联矩阵，顶点v_n为参考点。让我们也把A划分为$A=[A_{11}, A_{12}]$，这里A_{11}和A_{12}的列分别对应于RLC元件以及电流源。如果I_1和I_2表示RLC元件电流和电流源的列矢量，那么N的KCL方程变为

$$A_{11}I_1=-A_{12}I_2$$

此外有

$$I_1=Y_1V_1$$

这里V_1是RLC元件的电压列矢量，Y_1是相应的导纳矩阵。进而，根据节点变换式(11.12)则有

$$V_1=A_{11}^tV_n$$

这里V_n是节点电压列矢量。所以，根据KCL方程，可得到

$$(A_{11}Y_1A_{11}^t)V_n=-A_{12}I_2$$

上面的方程称为节点方程，矩阵$A_{11}Y_1A_{11}^t$称为N的节点-导纳矩阵。

11.3 混合-变量法

本节，我们将讨论网络分析的混合-变量法。在这个方法中，一些独立变量是电压，而另一些是电流。这种方法本质上是回路法和割集法的组合。我们仅讨论包含独立电压源和电流源的RLC网络（没有互感）。这个方法也可以直接扩展到包含互感的网络中。

考虑一个连通网络N。让我们把N的元件分成二个集合E_1和E_2，以使E_1包含所有电压源，E_2包含所有电流源。设N_1'是从N中移去E_2得到的网络，N_2^*是收缩E_1所有元件后所得到的网络。设T_1是N_1'的生成林，T_2是N_2^*的生成树。那么$T=T_1\cup T_2$就是N的生成树。我们选择T_1和T_2，以使T包含所有电压源、但不包含电流源。让我们定义

$T_v=T_1$的子图，它包含除电压源外T_1的所有元素。

$\bar{T}_1=N_1'$中T_1的补。

$\hat{T}_2=N_2^*$中T_2的补。

$T_i=\hat{T}_2$的子图，它包含除电流源外$\hat{T}_2$的所有元素。

让我们划分元件电压矢量V_e和元件电流矢量I_e如下：

$$V_e=\begin{bmatrix} V_g \\ V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_f \end{bmatrix}$$

$$I_e = \begin{bmatrix} I_E \\ I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_J \end{bmatrix}$$

这里下标E、J、1、2、3和4分别表示电压源、电流源、T_v、T_2、$\hat{T}_1$和T_l元件。

我们现通过变量V_t和I_l来对N进行描述。运用对应于T的B_f和Q_f矩阵，我们能写出KVL和KCL如下：

$$\begin{array}{c} \\ \\ 3 \\ 4 \\ J \end{array}\begin{array}{cccccc} & (T_v) & (T_2) & (\hat{T}_1) & (T_l) & \\ E & 1 & 2 & 3 & 4 & J \\ A & B & 0 & U & 0 & 0 \\ C & D & F & 0 & U & 0 \\ G & H & K & 0 & 0 & U \end{array} \begin{bmatrix} V_E \\ V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_J \end{bmatrix} = [0] \qquad (11.25)$$

（注意：请考虑对应于T_2的列中零子矩阵的存在。）

$$\begin{array}{c} \\ E \\ 1 \\ 2 \end{array}\begin{array}{cccccc} E & 1 & 2 & 3 & 4 & J \\ U & 0 & 0 & -A^t & -C^t & -G^t \\ 0 & U & 0 & -B^t & -D^t & -H^t \\ 0 & 0 & U & 0 & -F^t & -K^t \end{array} \begin{bmatrix} I_E \\ I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_J \end{bmatrix} = [0] \qquad (11.26)$$

考虑式（11.25）和（11.26）中的第二个方程组：

$$FV_2+V_4=-CV_E-DV_1 \qquad (11.27)$$

$$I_1-B^tI_3=D^tI_4+H^tI_J \qquad (11.28)$$

利用v-i关系

$$V_2=Z_2I_2$$
$$V_4=Z_4I_4$$
$$I_1=Y_1V_1$$
$$I_3=Y_3V_3$$

我们就能把式(11.27)和(11.28)写成

$$FZ_2I_2+Z_4I_4=-CV_E-DV_1 \qquad (11.29)$$

$$Y_1V_1-B^tY_3V_3=D^tI_4+H^tI_J \qquad (11.30)$$

根据回路和割集变换，则得

$$I_3=F^tI_4+K^tI_J \qquad (11.31)$$

$$V_3=-AV_E-BV_1 \qquad (11.32)$$

现把上式I_2和V_3代入式（11.29）和（11.30），并重新整理，则得

$$\begin{bmatrix} Z_4+FZ_2F^t & D \\ -D^t & Y_1+B^tY_3B \end{bmatrix}\begin{bmatrix} I_4 \\ V_1 \end{bmatrix}=\begin{bmatrix} -C & -FZ_2K^t \\ -B^tY_3A & H^t \end{bmatrix}\begin{bmatrix} V_g \\ I_j \end{bmatrix}$$

上式称为混合方程组或混合-变量方程组。

很容易证实，一旦用上式来确定I_4和V_1，那么所有其它变量就能够容易地用式(11-25)、(11.26)以及回路、割集变换计算出来。

注意：当我们选$E_1=\varnothing$和$E_2=E$，那么混合-变量法就变成了回路法；如选择$E_1=E$和$E_2=\varnothing$，那么混合-变量法就变成了割集法。这里E是N的元件集合。

进一步，混合变量方程组包含了$\rho(N_1^{\prime})+\mu(N_2^{*})-n_v-n_c$个变量。很显然，$\rho(N_1^{\prime})+\mu(N_2^{*})$的数目取决于$E_1$和$E_2$的选择。因此，引出了如何划分网络$N$的元件集合$E$为（$E_1$，$E_2$）并使得$\rho(N_1^{\prime})+\mu(N_2^{*})$尽可能小的问题。这种划分方法将在下节讨论。

11.4 图的主划分

本节所描述的图的主划分，是由Kishi和Kajitani在文献[11.7]中提出的。我们将会看到，一个图G的主划分被定义为当对G的边集E的划分(E_1,E_2)进行混合参量分析时，这种划分导致独立变量的数目最少。我们这里所进行的讨论是建立在文献[11.7]和[11.8]基础上的。

考虑一个连通图G。本节中所考虑的G的所有子图都是边-导出子图。因此，一个子图和它的边集将用同一个符号来表示。

G中任意两个生成树T_1和T_2之间距离$d(T_1, T_2)$被定义为

$$d(T_1,\ T_2)=|T_1-T_2|=|T_2-T_1|$$

因此，$d(T_1,T_2)$等于出现在$T_1(T_2)$中、但不出现在$T_2(T_1)$中的边数。我们可容易地证明。

$$d(T_1,\ T_2)=\rho(G)-\text{在}T_1\text{和}T_2\text{中公共树支的数目}$$
$$=\mu(G)-\text{在}T_1\text{和}T_2\text{中公共连支的数目}$$

如果对于所有G的生成树对T_i和T_j，都有$d(T_1,\ T_2)\geqslant d(T_i,\ T_j)$那么便称生成树$T_1$和$T_2$具有最大距离。

例如，$T_1=\{2,\ 3,\ 4,\ 7\}$和$T_2=\{1,\ 3,\ 5,\ 6\}$构成了图的最大距离生成树对，如图11.4所示

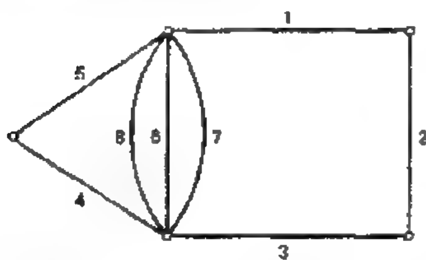


图 11.4

定理11.4 设T_1和T_2构成了连通图G的最大距离生成树对。

1.对应于T_1或T_2并且由T_1和T_2所公有的一连支所确定的G的基本回路不包含这些生成树所公有的树支。

2.对应于T_1或T_2并且由T_1和T_2所公有的一树支所确定的G的基本割集，不包含这些生成树所公有的连支。

证明

1.假设对应于T_1、并并由公共连支C所确定的G的基本回路包含了公共树支b。那么

在T_2和生成树$T_1' = (T_1 - b) \cup c$之间的距离$d(T_1', T_2)$，满足

$$d(T_1', T_2) = d(T_1, T_2) + 1$$

这与T_1和T_2具有最大距离相矛盾。

2. 用对偶法可证明之。 □

对于任何两个生成树T_1和T_2，设c是公共连支，b是公共树支。那么序列

$$P: cT_1 e_1 T_2 e_2 T_1 \cdots e_i T^* b$$

这里T^*不是T_1就是T_2。如果P具有下述性质，那么就称为从公共连支c到公共树支b，且长度为i的派生序列：

1. T_1和T_2在P中交替出现。

2. e_1是在对应于T_1、并且由c所确定的基本回路中。

3. b是在对应于T^*、并且由e_i所确定的基本回路中。

4. 如果e_j和e_{j+1}在P中以$e_j T_a e_{j+1}$出现，这里$T_a = T_1$或T_2，那么e_j就是T_a的一个连支，e_{j+1}就是T_a的一个树支，而且e_{j+1}是在对应于T_a、并且由e_j所确定的基本回路中。

在上述定义中，可用割集替代回路，连支替代树支，我们就能用完全对偶的方法定义一个从公共树支到公共连支的派生序列。事实上，如果P是从公共连支c到公共树支b的派生序列，那么序列P'将同倒序写出的P一样，是从b到c的派生序列。

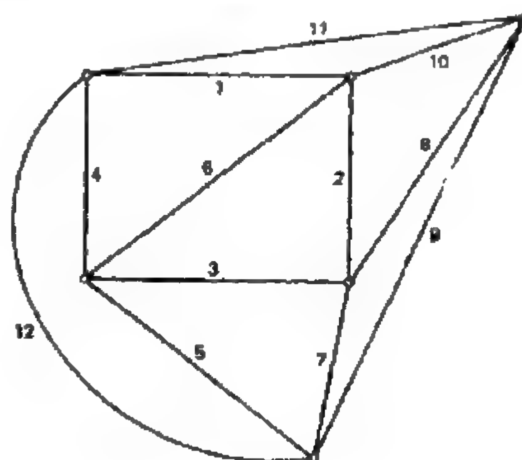


图 11.5

举例考察图11.5所示图的生成树$T_1 = \{1, 2, 3, 7, 10\}$和$T_2 = \{2, 6, 8, 11, 12\}$，图中边5是$T_1$和$T_2$的公共连支，边2是$T_1$和$T_2$的公共树支。因此

$$P: 5T_1 7T_2 8T_1 2$$

是从5到2的派生序列，而

$$P': 2T_1 8T_2 7T_1 5$$

是从2到5的派生序列。

令T_1和T_2构成了最大距离生成树对。假设存在一个从公共连支c到公共树支b的派生序列

$$P: cT_1 e_1 T_2 b$$

其长度为1。那么T_1和$T_2' = (T_2 - b) \cup e_1$将构成最大距离生成树对，其中$c$是公共连支，$e_1$是公共树支。进而，公共树支$e_1$将出现在对应$T_1$、并且由公共连支$C$所确定的基本回路中，这和定理11.4是相矛盾的。所以，不存在长度为1的、从c到b的派生序列。用对偶的方法，我们也能证明不存在长度为1的、从b到c的派生序列。

定理11.5 设c和b分别为一个连通图的生成树对T_1和T_2的公共连支和公共树支。

如果T_1和T_2具有最大距离，那么不存在从c到b的派生序列，也不存在从b到c的派生序列。

证明

证明是采用对任何最大距离生成树对的派生序列的长度进行归纳的方法。

如同我们刚才看到的那样，对于任何最大距离生成树对，都不存在长度为1、从公共连支到公共树支的派生序列。假设长度小于$k \geqslant 2$的这样序列不存在。

如果存在一个最大距离生成树对T_1和T_2的、长度为k、从公共连支c到公共树支b的派生序列

$$P: cT_1e_1T_2e_2T_1\cdots e_kT^*b$$

那么这个派生序列将是同类序列中最短的。因为根据假设不存在长度小于k的这样序列。

如果$T^*=T_1$，那么T_2和$T_1'=(T_1-b)\cup e_k$就构成了最大距离生成树对，其中c为其公共连支，e_k为其公共树支。因为P是长度为k的最短序列，则可证明

$$P': cT_1'e_1T_2e_2T_1'\cdots e_{k-1}T_2e_k$$

是从c到e_k的派生序列，但是P'的长度为$k-1$，这与假设产生了矛盾，因为我们假设不存在长度为$k-1$的派生序列。

同样，如果$T^*=T_2$，那么T_1和$T_2'=(T_2-b)\cup e_k$构成了最大距离生成树对，其中c是其公共连支，e_k为其公共树支。可以再次证明

$$P'': cT_1e_1T_2'e_2T_1\cdots e_{k-1}T_1e_k$$

是从c到e_k的派生序列。因为P''的长度为$k-1$，这也是一个与假设相矛盾的结果。

因此，任何长度的、从公共连支到公共树支的派生序列是不存在的。同样，任何长度的、从公共树支到公共连支的派生序列也是不存在的。 □

给定一对最大距离生成树T_1和T_2，假如c是T_1和T_2的公共连支，G对应的k-子图G_c，可用如下方法构成：

1.设L_1是对应于T_1、并且由c所确定的基本回路中的所有边的集合。根据定理11.4，L_1不存在公共树支。

2.设L_2是对应于T_2、并且由L_1中所有边所确定的所有基本回路的并。根据定理11.5，L_2中没有公共树支。

3.重复上述步骤，我们能够得到边集L_1、L_2、…的序列，直到集合$L_{K+1}=L_K$为止。那么在边集L_K上产生的诱导子图，称为对应于c的k-子图G_c。

在上述的进程中，用割集替代回路，树支替代连支，我们能够用完全对偶的方法，来确定一个对应于公共树支b的k-子图G_b。

对应于公共连支的主子图G_1是所有对应于公共连支的k-子图的并，对应于公共树支的主子图G_2是所有对应于公共树支的k-子图的并。

例如，在图11.4所示的图中，对应于生成树对$T_1=\{2, 3, 4, 7\}$与$T_2=\{1, 3, 5, 6\}$的主子图G_1和G_2为：

$$G_1=\{6, 7, 8\}$$
$$G_2=\{1, 2, 3\}$$

文献[11.7]已经证明主子图G_1和G_2没有公共边。因为如若它们有公共边，那么我们可以构造一个从公共连支到公共树支的派生序列。

因此，任何图G由三个子图构成：

G_1——对应于公共连支的主子图。

G_2——对应于公共树支的主子图。

G_0——$G-(G_1\cup G_2)$的子图。

图G的这个划分（G_0，G_1，G_2）称为G的主划分。

有趣的是，文献[11.7]中图G的主划分是唯一的，并且与用来构造这种划分的最大距离生成树无关。

主子图G_0，G_1和G_2的一些有用性质表述如下，它们是由这些子图的定义得到的。

P1. G_1含有所有公共连支，但无公共树支；G_2含有所有公共树支，但无公共连支。

P2. 对应于T_1或T_2、并且由G_1中任一条边所确定的任何基本回路，仅由G_1中的边构成。

P3. 对应于T_1或T_2、并且由G_2中任一条边所确定的任何基本割集，仅由G_2中的边构成。

P4. $T_1\cap G_1$和$T_2\cap G_1$都是G_1的生成林。

P5. $T_1\cap G_2$和$T_2\cap G_2$都是G_2'的生成树，其中G_2'是收缩那些不在G_2中的所有边而得到的。

P6. 任一边的端点都在G_1的相同片中，这个边也在G_1中。

P7. 任一边的端点都在$G_0\cup G_1$的相同片中，这个边也在$G_0\cup G_1$中。

P8. $T_1\cap G_0$和$T_2\cap G_0$都是图G_0'的生成林，其中G_0'是用收缩G_1中所有的边、并且移去G_2中所有边而得到的。（这个性质可利用性质3来证明）

注意，如果G没有公共连支，那么$G_1=\varnothing$；如果G没有公共树支，那么$G_2=\varnothing$。

由图G中边集E的划分所导出性质P1的结论，如图11.6所示。根据这些性质，可以得到

$$E_1=A\cup D\cup J$$
$$E_2=B\cup K\cup H$$
$$E_0=C\cup F$$

这里E_1、E_2和E_0分别表示G_1、G_2和G_0的边集。

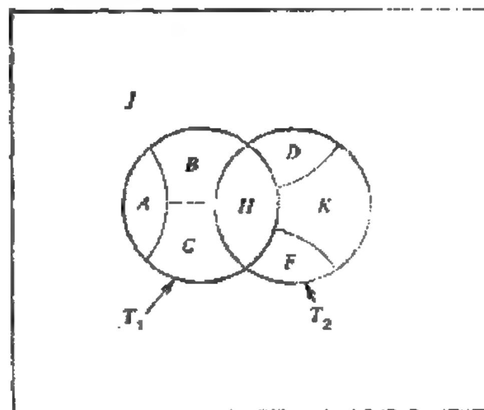


图 11.6

注意，作为性质P_4和P_5的结论，我们可得到$|A|=|D|$和$|B|=|K|$。

在上面讨论中引出的结果，导出了如下定理。

定理11.6 对于一个以（G_0，G_1，G_2）为其主划分的图G，

1. 如果$G_1\neq\varnothing$，那么$\rho(G_1)<\mu(G_1)$。

2.如果$G_2 \neq \varnothing$，那么$\rho(G_2') > \mu(G_2')$，这里G_2'是收缩了（$G_0 \cup G_1$）中所有边而得到的图。

3.$\rho(G_0') = \mu(G_0')$，这里G_0'是收缩了G_1中所有边、并且移去了G_2中所有边而得到的图。

4.任何两个生成树之间的最大距离为$\rho(G_1) + \mu(G_2') + \rho(G_0')$。

证明

1.根据性质P4和$|A| = |D|$这个事实，则得

$$\rho(G_1) = |A|$$

如果$G_1 \neq \varnothing$，

$$\rho(G_1) < |D| + |J| = \mu(G_1)$$

2.根据性质P5和$|B| = |K|$这个事实，则得

$$\rho(G_2') = |B| + |H|$$

如果$G_2 \neq \varnothing$，

$$\rho(G_2') > |K| = \mu(G_2')$$

3.根据性质P8和$|C| = |F|$这个事实，则得

$$\rho(G_0') = |C| = |F| = \mu(G_0')$$

4.G的任何两个生成树之间的最大距离是

$$|A| + |B| + |C| = \rho(G_1) + \mu(G_2') + \rho(G_0') \qquad \square$$

设d_m表示一个图G的两个生成树之间最大距离。设（G_0，G_1，G_2）是G的主划分，那么根据以上定理，则有

$$d_m = \rho(G_1) + \mu(G_2') + \rho(G_0')$$

但是$\mu(G_2') + \rho(G_0')$等于图（$G_0 \cup G_2$）$'$的零度，其中图$(G_0 \cup G_2)'$是收缩了G_1中所有边而得到的（见图11.6）。所以

$$d_m = \rho(G_1) + \mu((G_0 \cup G_2)')$$

设E_a和E_b构成了G的边集E的任一划分。设G_a是边集E_a的子图，G_b'是收缩了G_a中所有边而得到的图。文献[11.8]已经证明

$$d_m \leqslant \rho(G_a) + \mu(G_b')$$

曾记得，如果我们使用了E的划分（E_a，E_b），那么$\rho(G_a) + \mu(G_b')$就等于混合-变量分析中的独立变量数。所以，我们可知，d_m是混合-变量分析中所需独立变量的最小数。这个数可能小于G的秩和零度。因此，混合-变量分析中的独立变量数可能小于回路和割集分析中的独立变量数。例如，图11.4所示图的主划分为

$$G_0 = \{4, 5\}$$
$$G_1 = \{6, 7, 8\}$$
$$G_2 = \{1, 2, 3\}$$

可以证实

$$\rho(G_1)=1$$

$$\mu((G_0 \cup G_1)')=2^*$$

所以

$$d_m=3$$

因此，这个网络的混合-变量分析仅需要三个独立变量。而回路和割集方法都需要四个独立变量。

文献[11.8]中称d_m为一个网络的拓扑自由度。在文献[11.8]中讨论了具有$\rho(G_0)+\mu(G_0')=d_m$的这些划分（E_0，B_0）的有趣性质。

文献[11.7]以及[11.8]对图论都有重大的贡献。文献[11.9]讨论一种用来计算图的主划分的算法。文献[11.10]把主划分的概念扩展到了拟阵。

11.5 状态方程

在11.2到11.4节中，我们讨论了描述一个网络方程的回路、割集和混合-变量的方程组。在时域中，这些方程都是微积分方程。本节，我们将给出有关一阶微分方程（没有积分）对网络的描述。采用这种描述方法的原因之一，就是在数学文献中有大量关于求解这些方程的信息以及适用于某种情况解答的性质。进而，状态表示法更有普遍意义，因为它可用于时变和非线性网络。此外我们只限于讨论含有互感、独立电压源和独立电流源的特殊类型的RLC网络。

电网络N的状态方程是用电容电压和电感电流的导数做为变量而形成的。很显然，并非所有电容电压都可以被选做独立变量，这是因为在网络中可能存在仅有电容构成的回路。同样，并非所有电感电流都能被选做独立变量，这是因为可能存在仅由电感构成的割集。正像用回路和割集方法分析那样，推导状态方程的起点就是选择适当的生成树。我们特地选择这样的一个生成树，使得它具有：

1. 具有所有的电压源，但无电流源；
2. 具有最大可能的电容数目；
3. 具有最小可能的电感数目。

按上述要求选择的生成树，称为正规树。

给定一个网络N，设N_1为N的一个子集，N_1中包含所有的电压源和电容，设N_2是N的一个子集，N_2中包含所有的电压源、电容和电阻。N的一个正规树，可按下述选择：

1. 首先选择N_1的生成林T_1，并使得T_1中包含了所有的电压源。
2. 然后选择N_2的生成林T_2，并使得$T_1 \subseteq T_2$。
3. 按需要在T_2中加入电感以得到N的生成树T这样得到的树就是N的正规树。

举例，图11.7(a)示出的网络及其正规树，如图11.7(b)所示。

在选择了正规树之后，让我们把元件的电压矢量V_e和电流矢量I_e划分如下：

* 原文为$\mu(G_0 \cup G_1)'=2$。——译者注

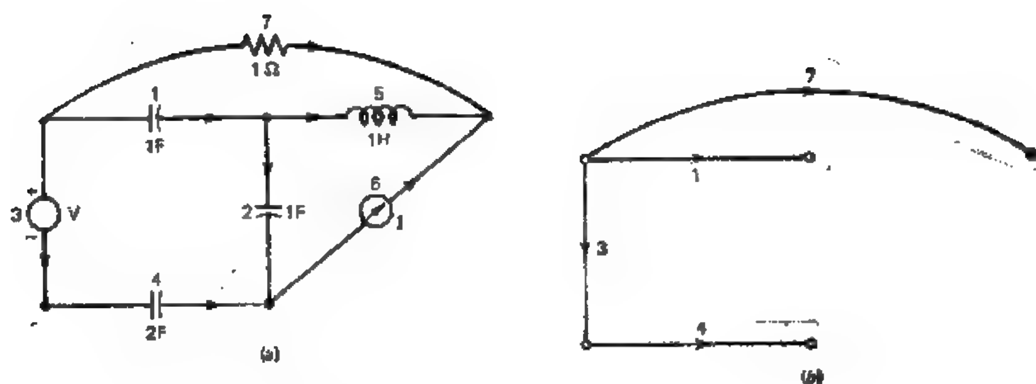


图 11.7 (a)网络；(b) (a)网络的正规树。

$$V_e=\begin{bmatrix} V_t \\ V_l \end{bmatrix} \text{ 和 } I_e=\begin{bmatrix} I_t \\ I_l \end{bmatrix}$$

并且

$$V_t=\begin{bmatrix} V_E \\ V_{Ct} \\ V_{Rt} \\ V_{Lt} \end{bmatrix}, \quad V_l=\begin{bmatrix} V_{Cl} \\ V_{Rl} \\ V_{Ll} \\ V_J \end{bmatrix}$$

$$I_t=\begin{bmatrix} I_E \\ I_{Ct} \\ I_{Rt} \\ I_{Lt} \end{bmatrix}, \quad V_l=\begin{bmatrix} I_{Cl} \\ I_{Rl} \\ I_{Ll} \\ I_J \end{bmatrix}$$

这里下标E、J、C、R和L分别表示电压源、电流源、电容、电阻和电感。下标t和l分别表示T的树支和连支。

基本割集矩阵$Q_f=[U \quad Q_{fl}]$的子矩阵Q_{fl}可划分为

$$Q_{fl}=\underset{\text{树支}}{\downarrow}\begin{matrix} \\ E \\ C \\ R \\ L \end{matrix}\overset{\longrightarrow\text{连支}}{\begin{matrix} C & R & L & J \\ \hline \end{matrix}}\begin{bmatrix} Q_{EC} & Q_{ER} & Q_{EL} & Q_{EJ} \\ Q_{CC} & Q_{CR} & Q_{CL} & Q_{CJ} \\ 0 & Q_{RR} & Q_{RL} & Q_{RJ} \\ 0 & 0 & Q_{LL} & Q_{LJ} \end{bmatrix} \tag{11.33}$$

注意：$Q_{RC}=0$。如若有个电容在对应于某个电阻的基本割集中，那么从T中移去这个电阻而加上这个电容，这将导致某个生成树比T多一个电容。因此，这将与正规树的选择相矛盾。用相同的方法，我们可以证明$Q_{LC}=0$，$Q_{LR}=0$。

根据KCL方程

$$Q_f I_e=0$$

因而得到：

$$I_E = -Q_{EC}I_{Cl} - Q_{ER}I_{Rl} - Q_{EL}I_{Ll} - Q_{EJ}I_J \qquad (11.34a)$$
$$I_{Ct} = -Q_{CC}I_{Cl} - Q_{CR}I_{Rl} - Q_{CL}I_{Ll} - Q_{CJ}I_J \qquad (11.34b)$$
$$I_{Rt} = \qquad -Q_{RR}I_{Rl} - Q_{RL}I_{Ll} - Q_{RJ}I_J \qquad (11.34c)$$
$$I_{Lt} = \qquad -Q_{LL}I_{Ll} - Q_{LJ}I_J \qquad (11.34d)$$

进而由割集变换得到

$$V_{Cl} = Q_{EC}^t V_E + Q_{CC}^t V_{Ct} \qquad (11.35a)$$
$$V_{Rl} = Q_{ER}^t V_E + Q_{CR}^t V_{Ct} + Q_{RR}^t V_{Rt} \qquad (11.35b)$$
$$V_{Ll} = Q_{EL}^t V_E + Q_{CL}^t V_{Ct} + Q_{RL}^t V_{Rt} + Q_{LL}^t V_{Lt} \qquad (11.35c)$$
$$V_J = Q_{EJ}^t V_E + Q_{CJ}^t V_{Ct} + Q_{RJ}^t V_{Rt} + Q_{LJ}^t V_{Lt} \qquad (11.35d)$$

运用元件的v-i关系和上述方程，我们得到以V_{Ct}和I_{Ll}为导数的状态方程。所有其它无源变量必须消去。

对于电容，不需要的变量是I_{Ct}、V_{Cl}和I_{Cl}。对于电感，不需要的变量是V_{Lt}、I_{Lt}和V_{Ll}。下面我们就消去这些变量。

首先重写式（11.34b）为

$$I_{Ct} + Q_{CC}I_{Cl} = [U \quad Q_{CC}]\begin{bmatrix} I_{Ct} \\ I_{Cl} \end{bmatrix} = -Q_{CR}I_{Rl} - Q_{CL}I_{Ll} - Q_{CJ}I_J$$

在上式中使用电容的v-i关系

$$\begin{bmatrix} I_{Ct} \\ I_{Cl} \end{bmatrix} = \begin{bmatrix} C_t & 0 \\ 0 & C_l \end{bmatrix} \frac{d}{dt} \begin{bmatrix} V_{Ct} \\ V_{Cl} \end{bmatrix}$$

并用式（11.35a）替代V_{Cl}，则得

$$\mathscr{C} \frac{d}{dt}(V_{Ct}) = -Q_{CR}I_{Rl} - Q_{CL}I_{Ll} - Q_{CJ}I_J + \mathscr{C}^* \frac{d}{dt}(V_E) \qquad (11.36)$$

这里

$$\mathscr{C} = [U \quad Q_{CC}]\begin{bmatrix} C_t & 0 \\ 0 & C_l \end{bmatrix}\begin{bmatrix} U \\ Q_{CC}^t \end{bmatrix}$$

$$\mathscr{C}^* = -[U \quad Q_{CC}]\begin{bmatrix} C_t & 0 \\ 0 & C_l \end{bmatrix}\begin{bmatrix} 0 \\ Q_{EC}^t \end{bmatrix}$$

注意：能够证明，$\mathscr{C}$就是从N中移去除电容以外的所有连支元件、并且收缩掉除电容以外的所有树支元件而得到的网络的割集导纳矩阵。

用相同的方法，从式（11.35c）出发，并且运用v-i关系，

$$\begin{bmatrix} V_{Ll} \\ V_{Lt} \end{bmatrix} = \begin{bmatrix} L_{ll} & L_{lt} \\ L_{tl} & L_{tt} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} I_{Ll} \\ I_{Lt} \end{bmatrix}$$

然后再用式（11.34d）替代I_{Lt}，对于电感则得到如下关系：

$$\mathscr{L} \frac{d}{dt}(I_{Ll}) = Q_{CL}^t V_{Ct} + Q_{RL}^t V_{Rt} + Q_{EL}^t V_E + \mathscr{L}^* \frac{d}{dt}(I_J) \qquad (11.37)$$

这里

$$\mathscr{L}=[U \quad -Q_{LL}^t]\begin{bmatrix} L_{tt} & L_{tt} \\ L_{tt} & L_{tt} \end{bmatrix}\begin{bmatrix} U \\ -Q_{LL} \end{bmatrix}$$

$$\mathscr{L}^*=[U \quad -Q_{LL}^t]\begin{bmatrix} L_{tt} & L_{tt} \\ L_{tt} & L_{tt} \end{bmatrix}\begin{bmatrix} 0 \\ -Q_{LJ} \end{bmatrix}$$

注意：能够证明，$\mathscr{L}$就是从N中移去除电感以外的所有连支元件、并且收缩掉除电感以外的所有树支元件而得到的网络的回路阻抗矩阵。

现在，我们必须根据式（11.36）和（11.37）中消去I_{Rl}和V_{Rt}。为此目的，考虑下面的v-i关系：

$$I_{Rl}=G_lV_{Rl}$$
$$V_{Rt}=R_tI_{Rt}$$

用式（11.35b）和（11.34c）来替代V_{Rl}和I_{Rt}之后，上面方程变为

$$I_{Rl}=G_lQ_{RR}^tV_{Rt}+G_lQ_{CR}^tV_{Ct}+G_lQ_{ER}^tV_E \tag{11.38}$$

$$V_{Rt}=-R_tQ_{RR}I_{Rl}-R_tQ_{RL}I_{Ll}-R_tQ_{RJ}I_J \tag{11.39}$$

用式（11.38）替代（11.39）中的I_{Rl}，使它成为

$$(U+R_tQ_{RR}G_lQ_{RR}^t)V_{Rt}=-R_tQ_{RR}G_lQ_{CR}^tV_{Ct}-R_tQ_{RR}G_lQ_{ER}^tV_E-R_tQ_{RL}I_{Ll}$$
$$-R_tQ_{RJ}I_J \tag{11.40}$$

由上面方程能解出V_{Rt}，当且仅当（$U+R_tQ_{RR}G_lQ_{RR}^t$）的逆矩阵存在。这个矩阵可写成R_tG形式，这里

$$G=G_t+Q_{RR}G_lQ_{RR}^t$$

矩阵$G_t+Q_{RR}G_lQ_{RR}^t$能被证明是从N中移去（除电阻以外）所有连支元件、并且收缩掉除了电阻的所有树支元件而得到的网络的割集导纳矩阵。因此，这个矩阵的逆矩阵存在。所以从式（11.40）能解出V_{Rt}。

下一步，我们用上面求出来的I_{Rl}和V_{Rt}，并代入式（11.36）和（11.37）中，则得如下状态方程：

$$\frac{d}{dt}\begin{bmatrix} \mathscr{C} & 0 \\ 0 & \mathscr{L} \end{bmatrix}\begin{bmatrix} V_{Ct} \\ I_{Ll} \end{bmatrix}=\begin{bmatrix} -\mathscr{Y} & \mathscr{H} \\ \mathscr{G} & -\mathscr{Z} \end{bmatrix}\begin{bmatrix} V_{Ct} \\ I_{Ll} \end{bmatrix}+\begin{bmatrix} -\mathscr{Y}^* & \mathscr{H}^* \\ \mathscr{G}^* & -\mathscr{Z}^* \end{bmatrix}\begin{bmatrix} V_E \\ I_J \end{bmatrix}$$
$$+\frac{d}{dt}\begin{bmatrix} \mathscr{C}^* & 0 \\ 0 & \mathscr{L}^* \end{bmatrix}\begin{bmatrix} V_E \\ I_J \end{bmatrix} \tag{11.41}$$

这里

$$R=R_l+Q_{RR}^tR_tQ_{RR}$$
$$\mathscr{Y}=Q_{CR}R^{-1}Q_{CR}^t, \qquad \mathscr{H}=-Q_{CL}+Q_{CR}R^{-1}Q_{RR}^tR_tQ_{RL}$$
$$\mathscr{Z}=Q_{RL}^tG^{-1}Q_{RL}, \qquad \mathscr{G}=Q_{CL}^t-Q_{RL}^tG^{-1}Q_{RR}G_lQ_{CR}^t=-\mathscr{H}^t \tag{11.42}$$

和

$$\mathscr{Y}^*=Q_{CR}R^{-1}Q_{ER} \qquad \mathscr{H}^*=-Q_{CJ}+Q_{CR}R^{-1}Q_{RR}^tR_tQ_{RJ}$$
$$\mathscr{Z}^*=Q_{RL}^tG^{-1}Q_{RJ} \qquad \mathscr{G}^*=Q_{EL}^t-Q_{RL}^tG^{-1}Q_{RR}G_lQ_{ER}^t \tag{11.43}$$

在RLC网络中，矩阵$\mathscr{C}$和$\mathscr{L}$都是非奇异的，因而我们能将式（11.41）重写为

$$\frac{d}{dt}\begin{bmatrix} V_{Ct} \\ I_{Lt} \end{bmatrix} = \mathscr{A}\begin{bmatrix} V_{Ct} \\ I_{Lt} \end{bmatrix} + \mathscr{B}\begin{bmatrix} V_E \\ I_J \end{bmatrix} + \mathscr{D}\frac{d}{dt}\begin{bmatrix} V_E \\ I_J \end{bmatrix} \tag{11.44}$$

一旦我们解式（11.44）求出V_{Ct}和I_{Lt}，便可以得出以V_{Ct}、I_{Lt}、V_E和I_J表示的所有其他电压和电流。文献[11.11]、[11.12]可给出更详细的结论。

因为式（11.44）存在对V_E和I_J求导，所以由式（11.44）给出的状态方程不是正规的形式，但是用适当地确定一组新变量的方法，式[11.44]能变为正规的形式。参见文献[11.11]。

举例，考虑图11.7(a)所示网络。这个网络的正规树如图11.7(b)所示。Q_{ft}矩阵可划分为如下形式：

$$Q_{ft} = \begin{array}{c} \\ 3 \\ 1 \\ 4 \\ 7 \end{array}\begin{array}{c|c|c} 2 & 5 & 6 \\ 1 & 0 & -1 \\ \hline -1 & -1 & 0 \\ 1 & 0 & -1 \\ \hline 0 & 1 & 1 \end{array}$$

由此得到

$$Q_{EC} = [\,1\,], \qquad Q_{RC} = [\,0\,]$$
$$Q_{EL} = [\,0\,], \qquad Q_{RL} = [\,1\,]$$
$$Q_{EJ} = [-1], \qquad Q_{RJ} = [\,1\,]$$
$$Q_{CG} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \qquad Q_{CL} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \qquad Q_{CJ} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

利用上述矩阵，则得

$$\mathscr{C} = \begin{bmatrix} 2 & -1 \\ -1 & 3 \end{bmatrix}, \qquad \mathscr{C}^* \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \qquad \mathscr{L}[\,1\,], \qquad \mathscr{L}^* = [\,0\,]$$

$$G = [\,1\,], \qquad R = [\,0\,]$$

$$\mathscr{Y} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathscr{Y}^* = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \mathscr{H} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \mathscr{H}^* = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\mathscr{P} = [\,1\,], \qquad \mathscr{P}^* = [\,1\,], \qquad \mathscr{G} = [-1 \;\; 0\,], \quad \mathscr{G}^* = [\,0\,]$$

代入式（11.41），图11.7所示网络的状态方程为：

$$\frac{d}{dt}\left(\begin{array}{cc|c} 2 & -1 & 0 \\ -1 & 3 & 0 \\ \hline 0 & 0 & 1 \end{array}\right)\left(\begin{array}{c} v_1 \\ v_4 \\ \hline i_6 \end{array}\right) = \left(\begin{array}{cc|c} 0 & 0 & 1 \\ 0 & 0 & 0 \\ \hline -1 & 0 & -1 \end{array}\right)\left(\begin{array}{c} v_1 \\ v_4 \\ \hline i_6 \end{array}\right) + \left(\begin{array}{c|c} 0 & 0 \\ 0 & 1 \\ \hline 0 & -1 \end{array}\right)\left(\begin{array}{c} v_3 \\ \hline i_5 \end{array}\right)$$

$$
+\begin{bmatrix} 1 & 0 \\ -1 & 0 \\ \hline 0 & 0 \end{bmatrix}\begin{bmatrix} \frac{dv_3}{dt} \\ \hline \frac{di_6}{dt} \end{bmatrix}
$$

11.6 电阻网络的无增益特性

我们在研究电网络时用一个有趣的图论应用来结束本章。

在网络理论中，众所周知，一个由电阻和电源构成的网络，任何电阻上的电压值都不能大于电源压降值的总和。同样，通过任何电阻的电流值都不能大于通过电源的电流值。电阻网络的这个性质称为无增益性。本节，我们给出无增益性的证明。这个证明是纯图论的，在文献[11.13]中给出。

曾记得，一个回路的所有边只等同于回路的方向标上方向，则称为有向回路。一个割集的所有边用等同于割集的方向标上方向，则称为有向割集。

无增益性证明是以 Minty 的弧着色引理（定理10.31）的特殊情况为基础的。即在有向图中，每条边或者在有向回路中，或者在有向割集中，但不会同时在两个之中。

定理11.7 给定一个由电源和（线性/非线性）正值电阻构成的网络，那么通过任意非零电压的电阻中的电流值不大于通过电源电流值的总和。

证明

设所有零电压的元件都用短路来消除掉。然后，选择元件的参考方向，使得所有元件电压为正。再考虑某个非零电压的电阻，含有这样电阻的有向回路是不存在的。如若这样的一个有向回路存在，那么绕回路一周所有电压之和将是非零的，违反基尔霍夫电压定律。所以，如前所述，只能存在包含上述考虑的电阻的有向割集。

设通过所考虑电阻中的电流为 i_0。选择一个有向割集包含这个电阻。设 R 为在这个割集中其它所有电阻的集合。再设 S 为在这个割集中所有电源的集合。在这个割集中应用基尔霍夫定律，则

$$
i_0+\sum_{k\in R} i_k+\sum_{s\in S}\pm i_s=0
$$

因为电阻为正，所以对于每个电阻 $v_i\geqslant 0$。又因为所有电压为正，所以在每个电阻中的电流是非负的，并且能够写出

$$
|i_0|+\sum_{k\in R}|i_k|+\sum_{s\in S}\pm i_s=0
$$

所以，

$$
|i_0|\leqslant\sum_{s\in S}\mp i_s\leqslant\sum_{s\in S}|i_s|
$$

因此定理得证。 □

下面的定理是上面定理的对偶。用完全对偶的方法可证明。

定理11.8 给定一个由电源和（线性/非线性）正值电阻构成的网络，那么任何电阻上电压降值不大于所有电源电压降值的总和。 □

文献[11.14]和[11.15]是早期讨论无增益性的论文。

11.7 进一步阅读

最先推荐阅读文献[11.16]关于用图论研究电网络的专著。此外还推荐文献[11.17]、[11.18]和[11.19]三本书。本章中有些题目的详细讨论，特别是扩展到更一般的网络的问题可参见文献[11.11]。

文献[11.20]、[11.21]介绍了建立在撕裂概念上的一种称为分裂法的网络分析方法。作者应用这个概念来研究某一类型网络。这种方法的起源基于张量分析概念。文献[11.22]、[11.23]以及[11.24]、[11.25]阐明了分裂法的概念，并且在促使电机工程师更多了解这种方法上做出了很大贡献。在文献[11.26]、[11.27]、[11.28]、[11.29]和[11.30]中，可以找到对于分裂法的其它介绍，还可参阅文献[11.31]。

最近文献[11.32]介绍了混合-变量分析的通用形式，并且把当前存在的形式（包括分裂法）做为特例，文献[11.33]也讨论了回路和割集分析法的计算有效性问题。他们指出选择最优分析模式，从而导出最稀疏的回路阻抗矩阵和最稀疏的割集导纳矩阵。

文献[11.34]使用图论概念，特别是特勒根定理和弧着色引理的特殊情况，导出一些非线性网络和非线性多端口电阻网络的性质。

对于RLC网络的状态方程，首先由文献[11.35]和[11.36]推导出来。接着，讨论一般有源网络状态方程建立的论文不断涌现。例如，可参看文献[11.37]、[11.38]和[11.39]。

文献[11.16]给出了包含独立源的RLC网络可解性的充要条件。利用它们的基本推导，另外一些人不断地讨论了一般网络可解性问题。例如，参见文献[11.37]和[11.40]。

对电网络拟阵理论的探讨是一个迅速发展起来的很有前景的研究领域。从图出发来定义电网络是做为这类问题的推广，文献[11.41]、[11.42][11.43]引出了广义网络——基于拟阵的网络——的概念，并建立了网络的某些性质。可以预期，对于广义网络的进一步研究，将给网络理论中若干未解决的经典问题，特别是多端口电阻网络的综合问题提供新见解。

近期文献[11.44]以王氏代数[11.45]、[11.46]为工具讨论了拟阵。文献[11.47]利用拟阵并的概念研究了网络可解性问题。文献[11.48]已将网络理论中三个基本问题的统一处理方法做了表述，从而证明了这些问题都可归结为确定两个适当选择的拟阵的并的问题。

网络理论仍然是数学问题的丰富源泉。文献[11.49]讨论了一些这样的问题。

11.8 习 题

11.1 对于以顶点v_r为参考点的树T，其路径矩阵$P=[p_{ij}]$定义如下：

如果支路j是在T中从v_r到v_i的唯一路径上，那么$p_{ij}=+1$或-1取决于支路的方向与路径方向一致，还是相反；否则$p_{ij}=0$。利用节点变换，证明$P=(A^{-1})^t$，这里A是以v_r为参考点的树T的关联矩阵，并且导出定理6.12。

11.2 把网孔作为独立回路，导出平面网络的回路方程组。

11.3 设N是平面网络。设平面网络$\hat{N}$构造如下：

(a) 图$\hat{N}$是图N的对偶。令e和e'分别是N和$\hat{N}$的对应元素。

（b） 如果e是R欧姆的电阻，那么e'就是$\frac{1}{R}$欧姆的电阻。

（c） 如果e是K法拉（亨利）的电容（电感），那么e'就是K亨利（法拉）的电感（电容）。

（d） 如果e是值为$g(t)$的电流（压）源，那么e'就是值为$g(t)$的电压（流）源。

确定$\hat{N}$中电流源和电压源的方向，以使N的回路方程（选网孔为独立回路）变成$\hat{N}$的节点方程，这里回路电流用节点电压来代替。假定N的所有网孔都是顺时针方向的。（上面所定义的网络N和$\hat{N}$叫做对偶网络。）

11.4 画出图11.8所示网络的对偶。

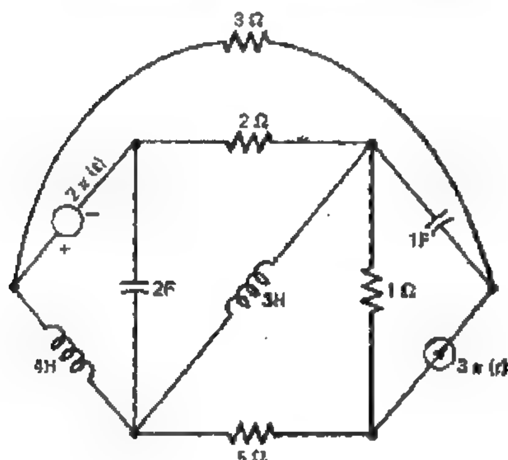


图 11.8

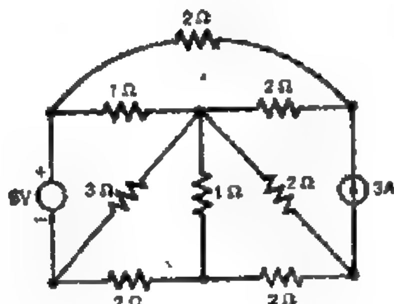


图 11.9

11.5 用回路分析法，确定图11.9所示网络的电压和电流。

11.6 用割集法和混合法，重做习题11.5。

11.7 求出图11.10所示图的主划分。

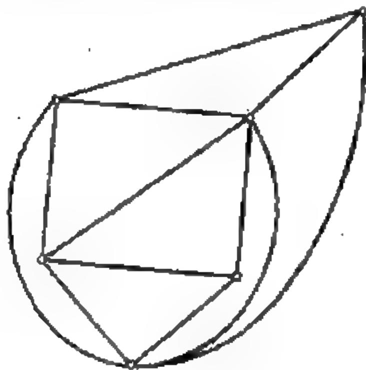


图 11.10

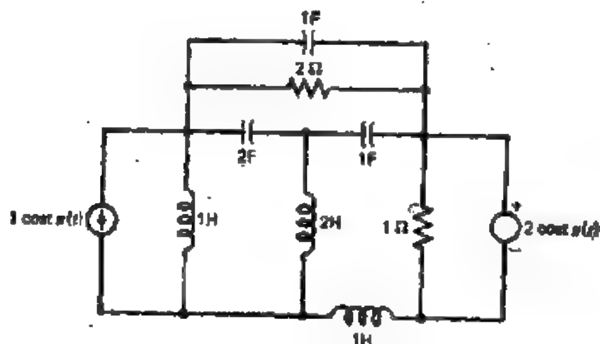


图 11.11

11.8 （a）导出图11.11所示网络的状态方程。

（b）用状态变量和电源变量来表示电阻上的电压和电流。

11.9 电网络N的复杂度是对于网络N所能给定初始条件的最大数目。它也等于动态元件上独立电流和电压的最大数目，这些独立元件上的电流和电压的瞬时值就足以确定网络的瞬时状态。进而，电网络N的复杂度也还等于N的自然频率数目。

对于RLC网络，证明：

（a） N的复杂度等于电抗元件的数目、减去线性独立全电感割集和线性独立全电容回路的数目之和。

（b） N的非零自然频率的数目等于N的复杂度减去线性独立全电容割集和线性独立全电感回路数目之和。

11.9 参考文献

11.1 B. D. H. Tellegen, "A General Network Theorem with Applications," *Philips Res. Rept.*, Vol. 7, 259—269(1952).

11.2 P. Penfield, Jr., R. Spence, and S. Duinker, "A Generalized Form of Tellegen's Theorem," *IEEE Trans. Circuit Theory*, Vol. CT-17, 302—305(1970).

11.3 P. Penfield, Jr., R. Spence, and S. Duinker, *Tellegen's Theorem and Electrical Networks*, M.I.T. Press, Cambridge, Mass., 1970.

11.4 J.L. Bordewijk, "Inter-Reciprocity Applied to Electrical Networks," *Appl Sci. Res.*, Vol. B6, 1—74(1956).

11.5 S. W. Director and R.A. Rohrer, "Automated Network Design——The Frequency Domain Case," *IEEE Trans. Circuit Theory*. Vol. CT-16, 330—337(1969).

11.6 G. Kishi and T. Kida, "Edge-Port Conservation in Networks," *IEEE Trans. Circuit Theory*, Vol. CT-15, 274—276(1968).

11.7 G. Kishi and Y. Kajitani, "Maximally Distant Trees and Principal Partition of a Linear Graph," *IEEE Trans. Circuit Theory*, Vol CT-16, 323—330(1969).

11.8 T. Ohtsuki, Y. Ishizaki, and H. Watanabe, "Topological Degrees of Freedom and Mixed Analysis of Electrical Networks," *IEEE Trans. Circuit Theory*, Vol. CT-17, 491—499(1970).

11.9 P.M. Lin, "An Improved Algorithm for Principal Parition of Graphs," *Proc. IEEE Intl. Symp. Circuits and Systems*, 1976, pp. 145—148.

11.10 J. Bruno and L. Weinberg, "The Principal Minors of a Matroid," *Linear Algebra and Its Appl.*, Vol. 4, 17—54(1971).

11.11 N. Balabanian and T. A. Bickart, *Electrical Network Theory*, Wiley, New York, 1969.

11.12 E.S. Kuh and R.A. Rohrer, "The State Variable Approach to Network Analysis," *Proc. IEEE*, Vol. 53, 672—686(1965).

11.13 D. H. Wolaver, "Proof in Graph Theory of the 'No-Gain' Property of Resistor Networks," *IEEE Trans. Circuit Theory*, Vol. CT-17, 436—437(1970).

11.14 A. Talbot, "Some Fundamental Properties of Networks without Mutual Inductance," *Proc. IEE(London)*, Vol. 102, 168—175(1955).

11.15 R. J. Schwartz, "A Note on the Transfer Ratio of Resistive Networks with Positive Elements," *Proc. IRE*, Vol. 43, 1670(1955).

11.16 S. Seshu and M. B. Reed, *Linear Graphs and Electrical Networks*, Addison-Wesley, Reading, Mass., 1961.

11.17 W. H. Kim and R. T. Chien, *Topological Analysis and Synthesis of Communi-*

cation Networks, Columbia Univ. Press New York, 1962.

11.18 W.K. Chen, *Applied Graph Theory*, North-Holland, Amsterdam, 1971.

11.19 W. Mayeda, *Graph Theory*, Wiley-Interscience, New York, 1970.

11.20 G. Kron,"A Set of Principles to Interconnect the Solution of Physical Systems,"*J. Appl. Phys.*, Vol. 24, 965—980(1953).

11.21 G.Kron, *Diakoptics: The Piecewise Solution of Large Scale Systems*, MacDonald, London 1963.

11.22 H. H Happ,"Diakoptics and Netwarhs, Acaelemic Rress,New York,1971.

11.23 H H. Happ,"Diakoptics——The Solution of System Problems by Tearing,"*Proc. IEEE*, Vol. 62, 930—940(1974).

11.24 F.H. Branin,"The Relation Between Kron's Method and the Classical Methods of Network Analysis," *Matrix and Tensor Quart*, Vol. 12, 69—115(1962).

11.25 F. H. Branin,"A Sparse Matrix Modification of Kron's Method of Piecewise Analysis," *Proc. IEEE Intl. Symp. Circuits and Systems*,1975, pp. 21—23.

11.26 K. Onodera,"Diakoptics and Codiakoptics of Electric Network," *RAAG Memoirs*, Vol.2, 369—388(1958).

11.27 J. Roth,"An Application of Algebraic Topology: Kron's Method of Tearing," *Quart. Appl. Math.*, Vol. 17, 1—24(1959).

11.28 S. Amari,"Topological Foundations of Kron's Tearing of Electical Networks," *RAAG Memoirs*, Vol. 3, 322—350(1962).

11.29 K.U Wang and T. Chao,"An Algebraic Theory of Network Topology," *Proc. IEEE Intl Symp. Circuits and Systems*, 1974,pp.324—328.

11.30 F.F Wu,"Solutions of Large-Scale Networks by Tearing",*IEEE Trans Circuits and Sys.*, Vol CAS-23, 706—713(1976).

11.31 L.O Chua and L.K Chen,"Nonlinear Diakoptics",*Proc. IEEE Intl. Symp Circuits and Systems*,1975,pp 373—376

11.32 L.O. Chua and L. K Chen, "Diakoptic and Generalized Hybrid Analysis," *IEEE Trans. Circuits and Sys.*, Vol. CAS-23, 694—705(1976).

11.33 L.O. Chua and L.K. Chen,"On Optimally Sparse Cycle and Coboundary Basis for a Linear Graph," *IEEE Trans. Circuit Theory*, Vol. CT-20, 495—503(1973).

11.34 L.O. Chua and D. N. Green,"Graph-Theoretic Properties of Dynamic Nonlinear Networks," *IEEE Trans. Circuits and Sys.*, Vol. CAS-23, 292—312(1976).

11.35 T. R. Bashkow,"The *A*-Matrix——A New Network Description,"*IRE Trans.Circuit Theory*, Vol. CT-4, 117—119(1957).

11.36 P.R. Bryant,"The Explicit Form of Bashkow's *A*-Matrix," *IRE Trans. Circuit Theory*, Vol. CT-9, 303—306(1962).

11.37 E.J Purslow,"Solvability and Analysis of Linear Active Networks by Use of the State Equations," *IEEE Trans Circuit Theory*, Vol. CT-17,469—475(1970).

11.38 O.Tosun and A. Dervisogtu,"Formulation of State Equations in Active RLC Networks," *IEEE Trans. Circuits and Sys.*, Vol. CAS-21, 36—38(1974).

11.39 S.K. Mark and M. N. S. Swamy, "The Generalized Tree for State Variables

in Linear Networks," *Int. J. Circuit Theory and Appl.*, Vol. 4, 87—92(1976).

11.40 W.K. Chen and F. N. Chan,"On the Unique Solvability of Linear Active Networks," *IEEE Trans. Circuits and Sys.*, Vol. CAS-21, 26—35(1974).

11.41 J.Bruno and L. Weinberg,"Generalized Networks: Networks Embedded on a Matroid —Part I," *Networks*, Vol. 6, 53—94(1976).

11.42 J. Bruno and L. Weinberg,"Generalized Networks: Networks Embedded on a Matroid —Part II," *Networks*, Vol. 6, 231—272(1976).

11.43 L. Weinberg,"Matroids, Generalized Networks, and Electric Network Synthesis," J. *Combinat. Theory B*, Vol. 23, 106—126(1977).

11.44 R.J. Duffin and T. D. Morley,"Wang Algebra and Matroids," *IEEE Trans Circuits and Sys.*, Vol. CAS-25, 755—762(1978)

11.45 R. J. Duffin, "An Analysis of the wang Algebra of Networks," *Trans. Am. Math. Soc.*, Vol 93, 114 -131(1959).

11.46 K. T. Wang,"On a New Method of Analysis of Electrical Networks," *Memoirs* 2, Nat. Res. Inst. Eng. Academia Sinica(1934).

11.47 B. Petersen,"Investigating Solvability and Complexity of Linear Active Networks by means of Matroids," *IEEE Trans. Circuits and Sys.*, Vol CAS-26, 330—342(1979)

11.48 M. Iri and N. Tomizawa,"A Unifying Approach to Fundamental Problems in Network Theory by means of Matroids," *Electron. and Commun. in Japan*, Vol. 58-A, 28—35(1975).

11.49 R.J. Duffin,"Electrical Network Models," in *Studies in Graph Theory, Part II*, The Mathematical Association of America, 1975, pp. 94—138.

第十二章 N端口电阻网络

12.1 引 言

如果一个网络具有n对可与外部器件如电流或电压源相连接的可进入的端点，则称这个网络是n端口网络。每一对这样可进人的端点称为一个端口。显然，一个n端口网络最多具有$2n$个不同的端点，在所有情况下，它的每个端口都有自己的一对端点；而它最少具有$n+1$个不同端点。当一个端点是所有端口的公共端时，这就是后种情况。一个n端口网络中，端口端点之外的顶点称为网络的内部顶点。n端口网络的每个端口都是与两个可变量相联系着的，即跨接端口两端的电压和通过端口的电流。图12.1给出了一个具有6个端点的3端口网络。端口电压和端口电流参考方向的选择，如图12.1所示。

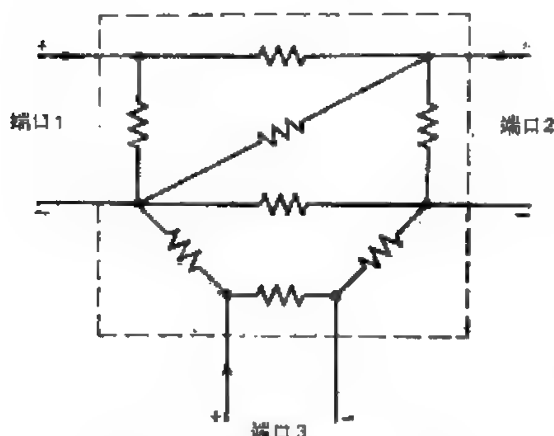


图 12.1 具有6个端点的3端口网络

在有向图表示中，n端口网络的每一个端口用一条边来表示，称之为连接相应端点的端口边。端口边可以看作代表跨接在这个端口上的外部器件。由标明每个端口端点正负参考方向后的所有端口边组成的子图，称为n端口网络的端口图。

端口边可以由端口的正参考端点指向负参考端点，或者方向相反。考虑图12.1中端口电压和端口电流的参考方向，则可很清楚地看出：在端口方向的前一种选择中，端口电压是和与端口边相联系的电压相等的，而端口电流是和与端口边相联系的电流负值相等的。端口方向按后一种选择时的对应关系也是显而易见的。

如果n端口网络的端口与外部器件相连接，则称之为增广n端口网络。

我们假定一个n端口网络的增广n端口网络是连通的，且在网络中不存在内部顶点。如果增广n端口网络是不连通的，那么增广网络的每个片都分别看作一个多端口网络。如果给定网络中存在内部顶点，那么我们就可以用星-三角变换来压缩它，依端口变量的关系来说，得到一个与原网络等效的网络。这样我们刚才所作的两个假定就不失一般性了。此外我们还不允许存在并行边（电导）。如果网络中存在并行边的话，那么可以把它合并成一条边。

一个n端口网络的特性，可以用任一组含有$2n$个端口变量的n个独立方程来描述。对于我们特别感兴趣的是，短路导纳和开路阻抗矩阵的描述。

在短路导纳矩阵的描述中，选择端口电压为独立变量。短路导纳矩阵$Y=[y_{ij}]$是一个

$n\times n$阶矩阵，它把端口电压矢量V_p变换为端口电流矢量I_p，因此有

$$YV_p=I_p \tag{12.1}$$

如果$V_p=[v_1,\ v_2,\ \cdots,\ v_n]^t$，$I_p=[i_1,\ i_2,\ \cdots,\ i_n]^t$，这里$v_j$和$i_j$分别是与第$j$个端口相联系的电压和电流变量，那么显然有

$$y_{kj}=\frac{i_k}{v_j}\bigg|_{v_i=0,\ \text{对于所有}\ i\neq j}$$

它的物理意义就是：当第j个端口被一个单位电压源激励而所有其它端口保持短路时，流过第k个端口的电流。

导纳矩阵存在的一个必要条件就是：端口图不具有回路，否则就不可能选择端口电压作为独立变量。

在n端口网络的开路阻抗矩阵的描述中，选择端口电流作为独立变量。开路阻抗矩阵$Z=[z_{ij}]$是一个$n\times n$阶矩阵，它把端口电流矢量I_p变换为端口电压矢量V_p，因此有

$$ZI_p=V_p \tag{12.2}$$

如果V_p和I_p的定义同前，那么有

$$z_{kj}=\frac{v_k}{i_j}\bigg|_{i_m=0,\ \text{对于所有}\ m\neq j}$$

它的物理意义就是：当端口j被一个单位电流源激励而所有其它端口保持开路时，第k个端口的两端电压。

如果上述阻抗矩阵存在的话，则必须在增广n端口网络中（已假定是连通的）没有仅由端口边所组成的割集，这是因为在此描述中，端口电流被选择为独立变量。换言之，阻抗矩阵的存在要求n端口网络是连通的。

下面推导RLC n端口网络N的Y矩阵和Z矩阵的表达式。我们用$\hat{N}$表示相应的增广网络。

我们首先推导N的Y矩阵。令T表示N的端口图。如前所述，我们要求T是无环的，以致每条端口边都可认为表示一个独立电压源，并假定每条端口边都是从其端口的正参考端点指向负参考端点。

令T_0表示$\hat{N}$的生成树，以使得T是T_0的一个子图。T的边称为端口树支，而T_0中剩余的边称为非端口树支。对应T_0的增广网络$\hat{N}$的基本割集矩阵可写成

$$\left[\begin{array}{c|c} Q_1 & U \\ \hline Q_2 & 0 \end{array}\right]$$

这里子矩阵

$$Q_t=\left[\frac{Q_1}{Q_2}\right]$$

对应网络N，Q_1的行对应端口树支，而Q_2的行对应非端口树支。

设列矢量V_p、V_{nb}和V_c定义如下：

V_p＝端口电压矢量

V_{nb}＝非端口树支电压矢量

V_e = N中边电压矢量

电流矢量I_p和I_e亦按同样的方式定义。令Y_e表示N中边导纳的对角线矩阵，以致

$$I_e = Y_e V_e \tag{12.3}$$

现在可以写出增广网络$\hat{N}$的独立KCL方程全式如下：

$$\left[\begin{array}{c|c} Q_1 & U \\ \hline Q_2 & 0 \end{array}\right]\left[\begin{array}{c} I_e \\ \hline -I_p \end{array}\right] = \left[\begin{array}{c} 0 \\ 0 \end{array}\right]$$

即

$$\left[\begin{array}{c} Q_1 \\ \hline Q_2 \end{array}\right][I_e] = \left[\begin{array}{c} I_p \\ \hline 0 \end{array}\right] \tag{12.4}$$

我们还可得出关系（割集变换，定理11.2）：

$$V_e = \left[\begin{array}{c} Q_1 \\ Q_2 \end{array}\right]^t \left[\begin{array}{c} V_p \\ \hline V_{nb} \end{array}\right] \tag{12.5}$$

将式（12.3）和（12.5）代入（12.4），可得

$$\left[\begin{array}{c|c} Y_{11} & Y_{12} \\ \hline Y_{21} & Y_{22} \end{array}\right]\left[\begin{array}{c} V_p \\ \hline V_{nb} \end{array}\right] = \left[\begin{array}{c} I_p \\ \hline 0 \end{array}\right] \tag{12.6}$$

这里

$$Y_{11} = Q_1 Y_e Q_1^t$$
$$Y_{12} = Q_1 Y_e Q_2^t = Y_{21}^t$$
$$Y_{22} = Q_2 Y_e Q_2^t$$

矩阵

$$Y_c = Q_f Y_e Q_f^t = \left[\begin{array}{c|c} Y_{11} & Y_{12} \\ \hline Y_{21} & Y_{22} \end{array}\right]$$

称为n端口网络N对应于T_0的割集导纳矩阵。

解出式（12.6）的第二组方程，得到

$$V_{nb} = -Y_{22}^{-1} Y_{21} V_p \tag{12.7}$$

注意Y_{22}是非奇异的，因为Q_2具有最大秩，而Y_e是非零对角线元素的对角矩阵。

将式（12.7）代入（12.6）的第一组方程，即

$$Y_{11} V_p + Y_{12} V_{nb} = I_p$$

则得

$$(Y_{11} - Y_{12} Y_{22}^{-1} Y_{21}) V_p = I_p \tag{12.8}$$

这样，N的短路导纳矩阵Y可由下式给出：

$$Y = Y_{11} - Y_{12} Y_{22}^{-1} Y_{21} \tag{12.9}$$

注意：如果端口图T具有$p>1$个片，那么网络N将具有$n+p$个端口端点。如果$p=1$，那么T将是$\hat{N}$的一个生成树。在这种情况下，$T = T_0$，即不存在非端口树支，并且这时的网络N称作秩为n的n端口网络。我们可以看出：一个秩为n的n端口网络的短路导纳矩阵Y与它的割集导纳矩阵Y_c是相等的，因此对于这种网络

$$Y = Q_f Y_e Q_f^t \tag{12.10}$$

注意：虽然事实上Q_f是N对应T_0的基本割集矩阵的子矩阵，但在一般情况下，Q_f不是$\hat{N}$

的基本割集矩阵。然而将发现，把Q_f称为N对应T_0的基本割集矩阵是很方便的。如果我们承认存在有零导纳的边的话，那么矩阵Q_f实际就是N的一个基本割集矩阵。

下面我们考虑RLC n端口网络N的Z矩阵的推导。如前所述，我们要求网络N是连通的，这样端口边就可由$\hat{N}$的某个生成树T_0的生成补树部分组成。现在的端口边称为端口连支，而生成补树中剩余的边称为非端口连支。在这种情况下，我们可以把每条端口边看作代表一个独立的电流源，并且从端口的负参考点指向正参考端点。则$\hat{N}$对应T_0的基本回路矩阵可写成：

$$\left[\begin{array}{c|c} B_1 & U \\ \hline B_2 & 0 \end{array}\right]$$

这里子矩阵

$$B_f=\left[\begin{array}{c} B_1 \\ \hline B_2 \end{array}\right]$$

对应于n端口网络N，其中B_1的行对应端口连支，而B_2的行对应非端口连支。

设V_p、V_e、I_p和I_e的定义同前。此外，令I_{ne}表示非端口连支电流向量。现在可以写出一组完全独立的KVL方程如下：

$$\left[\begin{array}{c|c} B_1 & U \\ \hline B_2 & 0 \end{array}\right]\left[\begin{array}{c} V_e \\ \hline -V_p \end{array}\right]=\left[\begin{array}{c} 0 \\ \hline 0 \end{array}\right] \tag{12.11}$$

如果Z_e是N的边阻抗对角矩阵，那么

$$V_e=Z_eI_e \tag{12.12}$$

我们还可得出关系（回路变换，定理11.2）

$$I_e=\left[\begin{array}{c} B_1 \\ \hline B_2 \end{array}\right]^t\left[\begin{array}{c} I_p \\ \hline I_{ne} \end{array}\right] \tag{12.13}$$

利用（12.11）、（12.12）和（12.13），我们得到

$$\left[\begin{array}{c|c} Z_{11} & Z_{12} \\ \hline Z_{21} & Z_{22} \end{array}\right]\left[\begin{array}{c} I_p \\ \hline I_{ne} \end{array}\right]=\left[\begin{array}{c} V_p \\ \hline 0 \end{array}\right] \tag{12.14}$$

这里

$$Z_{11}=B_1Z_eB_1^t$$
$$Z_{12}=B_1Z_eB_2^t=Z_{21}^t$$
$$Z_{22}=B_2Z_eB_2^t$$

解出式（12.4）中的第二组方程，得到

$$I_{ne}=-Z_{22}^{-1}Z_{12}I_p \tag{12.15}$$

我们可能注意到了：Z_{22}是非奇异的，因为B_2具有最大秩，而Z_e是对角线上没有零元素的对角矩阵。

最后，将式（12.15）代入（12.14）的第一组方程，得到

$$(Z_{11}-Z_{12}Z_{22}^{-1}Z_{21})I_p=V_p \tag{12.16}$$

这样，N的开路阻抗矩阵Z就由下式给出：

$$Z=Z_{11}-Z_{12}Z_{22}^{-1}Z_{21} \tag{12.17}$$

如果网络N没有回路，那么它是$\hat{N}$的一个生成树，因此，端口边将形成相应的生成补树。在这种情况下，将不存在非端口连支，N则称为零度为n的n端口网络。所以，一个零度为n的n端口网络的开路阻抗矩阵Z为

$$Z=B_f Z_e B_f^t \tag{12.18}$$

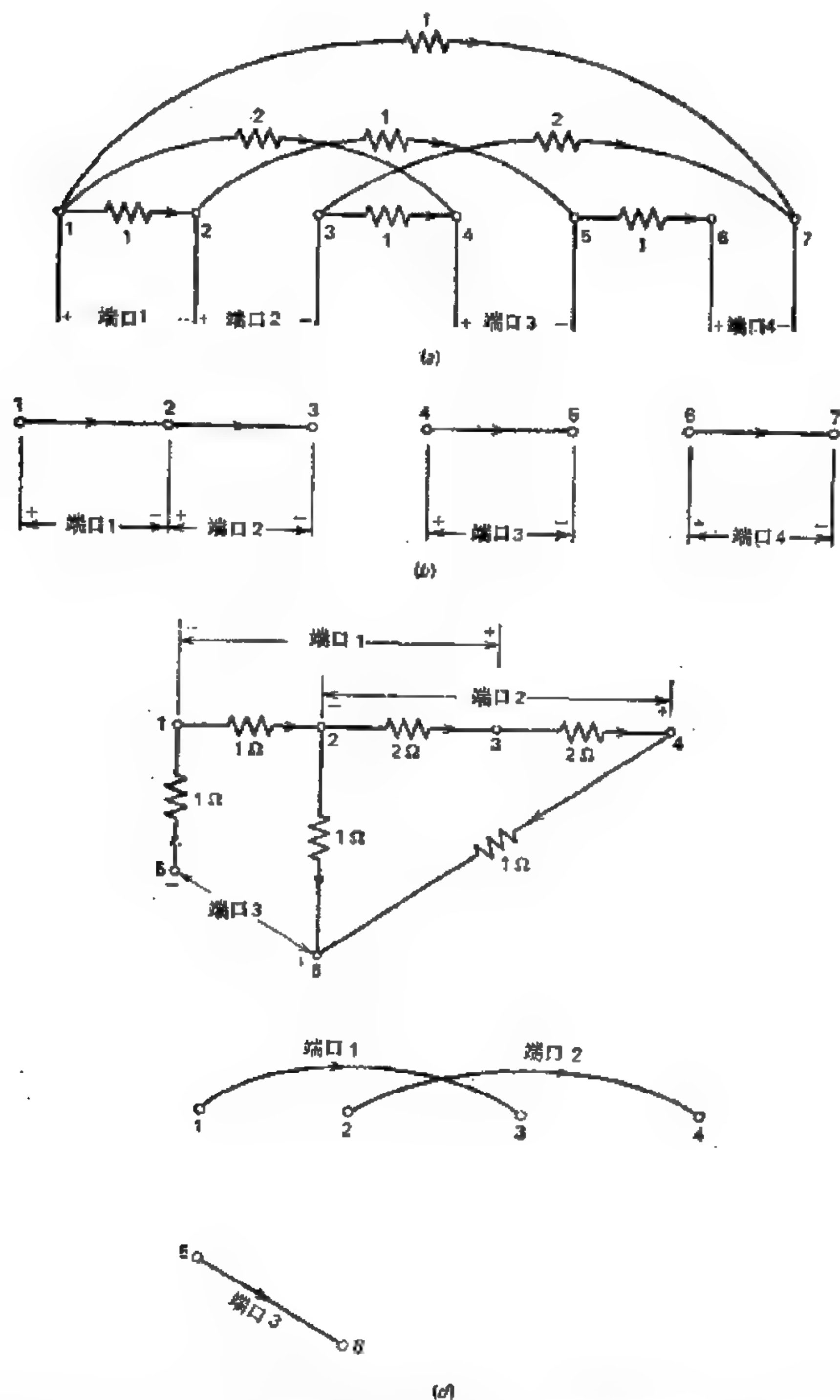


图 12.2 (a)电阻4端口网络(所有电导单位为西门子)；(b)图(a)网络的端口图；(c)电阻3端口网络；(d)图(c)网络的端口图。

这与Q_f矩阵的情形一样，，我们称B_f为网络N的一个基本回路矩阵。

下面举例说明n端口网络Y矩阵和Z矩阵的推导。

首先我们计算图12.2（a）所示的电阻4端口网络的Y-矩阵。这个网络的端口图，如图12.2（b）所示。选取边(3,4)和(5,6)为非端口树支，我们得到Q_f

$$Q_f=\left[\frac{Q_1}{Q_2}\right]$$

	(1,2)	(3,4)	(5,6)	(1,4)	(2,5)	(3,7)	(1,7)
端口 1	1	0	0	1	0	0	1
端口 2	0	0	0	1	1	0	1
端口 3	0	0	0	0	1	1	1
端口 4	0	0	0	0	0	1	1
非端口树支(3,4)	0	1	0	1	1	1	1
非端口树支(5,6)	0	0	1	0	0	1	1

对于该网络

$$\begin{aligned}Y_e&=\mathrm{diag}[g_{12}\quad g_{34}\quad g_{56}\quad g_{14}\quad g_{25}\quad g_{37}\quad g_{17}]\\&=\mathrm{diag}[\,1\quad 1\quad 1\quad 2\quad 1\quad 2\quad 1\,]\end{aligned}$$

这里g_{ij}称为连接顶点i和j的边（i，j）的导纳。我们得到

$$Q_fY_eQ_f^t=\begin{bmatrix}Y_{11} & Y_{12}\\ Y_{21} & Y_{22}\end{bmatrix}$$

$$=\left[\begin{array}{cccc|cc}4 & 3 & 1 & 1 & 3 & 1\\ 3 & 4 & 2 & 1 & 4 & 1\\ 1 & 2 & 4 & 3 & 4 & 3\\ 1 & 1 & 3 & 3 & 3 & 3\\ \hline 3 & 4 & 4 & 3 & 7 & 3\\ 1 & 1 & 3 & 3 & 3 & 4\end{array}\right]$$

则图12.2(a)所示的4端口网络的短路导纳矩阵为

$$Y=Y_{11}-Y_{12}Y_{22}^{-1}Y_{21}$$

$$=\frac{1}{19}\begin{bmatrix}51 & 23 & -11 & -2\\ 23 & 29 & 1 & -5\\ -11 & 1 & 21 & 9\\ -2 & -5 & 9 & 12\end{bmatrix}$$

下面考虑图12.2(c)所示的电阻 3 端口网络。该网络的端口图，如图12.2(d)所示。选取边(4，6)为非端口连支，则我们得到矩阵B_f为

$$B_f=\left[\frac{B_1}{B_2}\right]$$

$$
= \begin{array}{l} \text{端口 1} \\ \text{端口 2} \\ \text{端口 3} \\ \text{非端口连支(4, 6)} \end{array}
\overset{\begin{array}{cccccc}(1,2) & (2,3) & (3,4) & (1,5) & (2,6) & (4,6)\end{array}}{\left[\begin{array}{cccccc}
-1 & -1 & 0 & 0 & 0 & 0 \\
0 & -1 & -1 & 0 & 0 & 0 \\
-1 & 0 & 0 & -1 & -1 & 0 \\
\hline
0 & 1 & 1 & 0 & -1 & 1
\end{array}\right]}
$$

该网络的矩阵Z_e为

$$
\begin{aligned}
Z_e &= \operatorname{diag}[r_{12} \quad r_{23} \quad r_{34} \quad r_{15} \quad r_{26} \quad r_{46}] \\
&= \operatorname{diag}[\,1 \quad 2 \quad 2 \quad 1 \quad 1 \quad 1\,]
\end{aligned}
$$

现在得到

$$
B_f Z_e B_f^t = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix}
$$

$$
= \left[\begin{array}{ccc|c}
3 & 2 & 1 & -2 \\
2 & 4 & 0 & -4 \\
1 & 0 & 3 & 1 \\
\hline
-2 & -4 & 1 & 6
\end{array}\right]
$$

则网络的开路阻抗矩阵为

$$
\begin{aligned}
Z &= Z_{11} - Z_{12} Z_{22}^{-1} Z_{21} \\
&= \frac{1}{6} \begin{bmatrix} 14 & 4 & 8 \\ 4 & 8 & 4 \\ 8 & 4 & 17 \end{bmatrix}
\end{aligned}
$$

正如我们由式(12.10)和(12.18)所看到的，一个秩为n的n端口网络的Y矩阵的特性以及一个零度为n的n端口网络的Z矩阵的特性与它们相应的基本割集矩阵和基本回路矩阵是密切相关的。本章的后面几节将建立这些矩阵的一些性质，并讨论实现的步骤。

12.2 秩为n的电阻n端口网络的Y矩阵

本节，我们将讨论秩为n的电阻n端口网络的Y矩阵的几个重要性质。显然，这些n端口网络具有$n+1$个节点，所以，我们称之为$(n+1)$节点的n端口网络。在我们的所有讨论中，假定考虑中的网络不存在负电导。

12.2.1 基 本 性 质

考虑一个端口图为T的$(n+1)$-节点的电阻n端口网络N。由式(12.10)知道，N的短路电导矩阵Y为

$$
Y = Q_f G_e Q_f^t \tag{12.19}
$$

这里Q_f是N对应T的基本割集矩阵，G_e是N的边电导对角矩阵。

例如，对于图12.3所示的6端口网络，

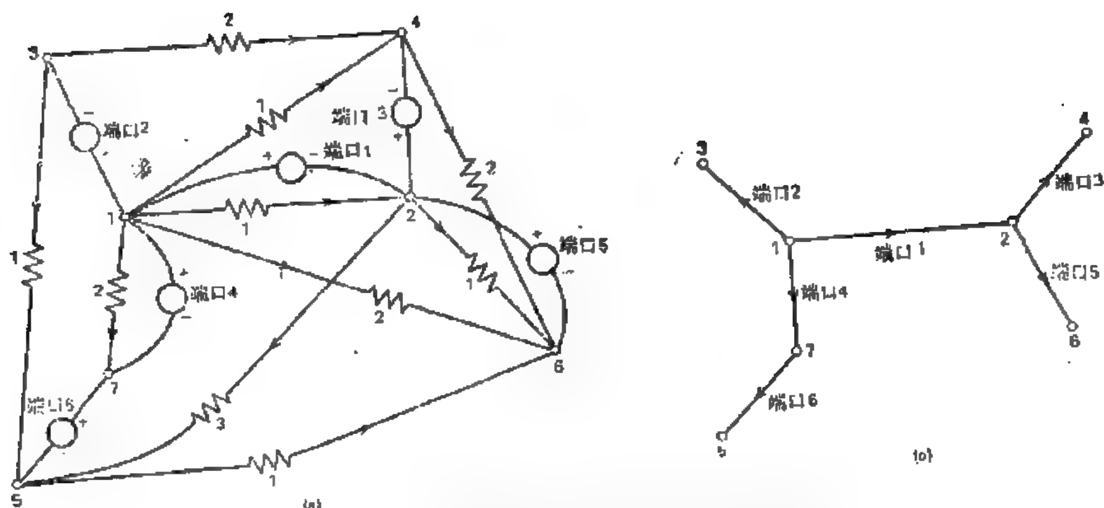


图 12.3 6端口网络和它的端口图(所有电导单位均为西门子)

$$
\begin{array}{c} \\ Q_f= \end{array}
\begin{array}{cccccccccc}
(1,2) & (1,4) & (1,6) & (1,7) & (2,5) & (2,6) & (3,4) & (3,5) & (4,6) & (5,6) \\
\end{array}
$$

$$
Q_f=\begin{bmatrix}
1 & 1 & 1 & 0 & -1 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & -1 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & -1
\end{bmatrix}
$$

以及

$$
G_e=\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
$$

因此，对于这个网络

$$
Y=Q_fG_eQ_f^t
=\begin{bmatrix}
10 & -2 & 3 & -4 & 3 & -4 \\
-2 & 3 & -2 & -1 & 0 & -1 \\
3 & -2 & 5 & 0 & -2 & 0 \\
-4 & -1 & 0 & 7 & -1 & 5 \\
3 & 0 & -2 & -1 & 6 & -1 \\
-4 & -1 & 0 & 5 & -1 & 5
\end{bmatrix}
\qquad (12.20)
$$

在下面的讨论中，我们用g_{ij}表示连接顶点 i 和 j 的边(i, j)的电导。我们假定Q_f中的第 i 行对应于端口 i 。为了进一步讨论，我们需要如下几个定义。

如果端口位于T中从顶点 i 到 j 的唯一路径中，则称电导g_{ij}生成N的一个端口。例如，在图12.3所示的网络中，电导g_{35}生成了端口2、4和6，注意：如果一个电导能够和一些端口一起构成一个基本回路，那么这个电导就可生成所有这些端口。

在网络理论的著作中，T的一个悬挂顶点通常称为末顶点。同样，T的一个悬挂端口称为末端口。所以在本章的所有讨论中，我们用术语“末顶点”和“末端口”代替“悬挂顶点”和“悬挂端口”，而且，我们用“短路”代替“收缩”。

一个树是线性树，如果它恰好有两个末顶点。换言之，一个线性树是一条路径。例如，图12.3中的端口1、2和5形成一个线性树。

一个树是星树，如果它所有的边都有一个公共的端顶点，这个端顶点称为星顶点。换言之，一个星树只有一个非末顶点。例如，图12.3中的端口1、2和4形成一个星树。

两个端口 i 和 j 称作是相同指向的，如果在含有这些端口的T的任意路径中，它们都指向相同的方向。否则它们就是相反指向的。

现在我们考虑矩阵Y的对角线上元素y_{ii}。由定理2.15可知，Q_f的第 i 行上的非零元素对应于所有生成端口 i 的那些电导。所以，由式（12.19）得到

$$y_{ii}=\text{所有生成端口}i\text{的电导之和} \qquad (12.21)$$

下面考虑Y的任意非对角线元素y_{ij}。令q_{ir}表示Q_f的(i,r)元素，那么我们知道（习题6.1）：如果对于任意 i 和 j 以及任意 r 和 s，乘积$q_{ir}q_{jr}$和$q_{is}q_{js}$都不为零的话，则它们具有相同的符号。因此，由式（12.19）得到

$$y_{ij}=\pm\{\text{所有既生成端口}\ i\ \text{又生成端口}\ j\ \text{的电导之和}\} \qquad (12.22)$$

现在我们来确定y_{ij}的符号。假定对于某个r，乘积$q_{ir}q_{jr}\neq 0$，则y_{ij}的符号与$q_{ir}q_{jr}$的符号相同。

由于q_{ir}和q_{jr}都是非零的，则显然端口 i 和 j 都在由对应Q_f中第r列的电导所产生的基本回路C上。因此出现了两种情况。

首先假定端口 i 和 j 具有相同指向。则端口 i 和 j 的指向或者与回路C的指向一致，或者相反，换言之，q_{ir}和q_{jr}将具有相同的符号。因此在这种情况下，y_{ij}是正的。

另一方面，如果端口 i 和 j 是相反指向的，那么端口 i 和 j 在回路C中将具有相反的指向，因而q_{ir}和q_{jr}具有相反的符号。因此，在这种情况下，y_{ij}是负的。这样，我们就有如下定理。

定理12.1 如果端口 i 和 j 是相同指向的，那么y_{ij}为正；否则y_{ij}为负。 □

由定理12.1很容易得出下面的结论。

定理12.2

1.如果端口 i 、j 和 k 在所有其它端口短路的情况下形成一个线性树，那么

$$y_{ij}y_{ik}y_{jk}\geq 0$$

2.如果端口 i 、j 和 k 在所有其它端口短路的情况下形成一个星树，那么

$$y_{ij}y_{ik}y_{jk}\leq 0$$ □

我们可以用图12.3所示的6端口网络的Y矩阵来验证定理12.1和定理12.2。

定理12.3 设端口i、j和k在所有其它端口短路的情况下形成一个线性树，进而假定在这个线性树中，端口以i、j和k的顺序出现，则

$$|y_{ij}| \geqslant |y_{ik}| \qquad \square$$

上面的结论产生于这样一个事实，即生成端口i和k的每一个电导，也一定生成端口j。定理12.3的一个重要推论叙述如下。

推论12.3.1 设y_{ij}是Y的一个非零元素，且具有最小数值。则一定存在一个电导生成端口i和j，这个电导的值就是$|y_{ij}|$。换言之，Q_f的i行和j行仅仅在一列上同时具有非零元素。（注意，假定并行边都已合并为一条边。） $\square$

例如，在式（12.20）的矩阵中，y_{24}是一个具有最小数值的非零元素，而g_{35}(图12.3)是一个生成端口2和4的电导。

定理12.4 设Y中第i行元素的最大数值为M_i，并且最大数值M_i的元素同时只出现在这一行的j、k、l、…列上。那么则由端口i、j、k、l…、所组成的子图是连通的。

证明

如果端口集合$\{i,j,k,l,\cdots\}$所形成的子图是不连通的，那么在这个集合中存在某个端口d，它不出现在含有端口i的片中。因为T是连通的，所以在T中存在端口p，p不在集合$\{i,j,k,l,\cdots\}$中，而且当T中的其它端口短路时，端口d、p、i形成一个出这三个端口依次出现的线性树，由定理12.3，我们注意到：$|y_{ip}| \geqslant |y_{id}|$。因为$y_{id}$是$Y$中第$i$行的最大值元素，所以它满足$|y_{ip}| = |y_{id}|$。因此，$p$也应当是集合$\{i,j,k,l,\cdots\}$中的一个元素，这就产生了矛盾，所以端口$i$、$j$、$k$、$l$、…组成的子图是连通的。 $\square$

定理12.5 设Y中第i行的最小元素为m_i，这里m_i可能为零。再设这一行的最小值元素同时出现在p、q、r、…列上，且仅出现在这些列上。那么由T的所有非p、q、r、…端口所组成的子图是连通的。

证明

如果定理中所说子图是不连通的，则存在一个端口j在集合$\{p, q, r, \cdots\}$中，而端口k不在$\{p, q, r, \cdots\}$中，并使得当T的所有其它端口短路时，端口i、j、k形成一个线性树，然后由定理12.3有：$|y_{ij}| \geqslant |y_{ik}|$，由于$|y_{ij}| = m_i$，即是$Y$中$i$行的最小元素，所以有$|y_{ik}| = m_i$，并且$k$在$\{p,q,r,\cdots\}$中，这就产生了矛盾。 $\square$

我们可以用式（12.20）的矩阵Y来验证上面两条定理。

由式（12.21）和（12.22）可以看出：对于所有i和j，$y_{ii} \geqslant |y_{ij}|$。这仅是由$Y$矩阵所呈现的更一般性质（称作最高性）的一种特例。

如果一个实对称矩阵的每个主子式不小于由同行（列）所构成的任意其它子式的值，那么就称这个矩阵是最高的。

文献[12.1]已经证明：任何矩阵$P = KDK^t$是最高的，这里K是单位模矩阵，而D是具有正实元素的对角矩阵。因为矩阵Q_f是单位模的，所以由（12.19）式可以得出结论：一个（$n+1$）节点的电阻n端口网络的短路电导矩阵是最高的。事实上，我们也可以证明：

任何n端口矩阵的短路电导和开路电阻矩阵都是最高的。

例如，考虑式（12.20）式中矩阵Y的子式

$$M_1=\begin{vmatrix}10 & -2 & 3\\ -2 & 3 & -2\\ 3 & -2 & 5\end{vmatrix} \text{和} M_2=\begin{vmatrix}10 & 3 & -4\\ -2 & -2 & -1\\ 3 & 5 & 0\end{vmatrix}$$

这两个子式都是由矩阵Y的前三行构成的，而前者是主子式。现在可以验证：$M_1=87$和$M_2=57$，因此$M_1\geq|M_2|$。

有关最高矩阵的详细讨论，参看文献[12.2]。

下面我们将讨论具有星树或线性树端口图的（$n+1$）节点电阻n端口网络Y矩阵的特性。我们说矩阵Y是以某种特定形式可存在的，如果Y能够通过一次或几次应用如下的运算而产生出那种形式：

1. 交换任意两行和相应的两列。
2. 改变任意一行和相应的列上的所有元素的符号。

注意：这两种运算分别对应于端口的重新编号和改变某些端口的指向。

12.2.2 星树端口图

考虑一个只有星树端口图T的（$n+1$）节点n端口网络N的Y矩阵，假定T中所有端口都指向星顶点（见图12.4），设星顶点编号为0，其它顶点分别为1、2、…、n、这样顶点i和0就组成了端口i的两个端点。由式（12.21）则有

$$y_{ii}=\sum_{\substack{j=1\\ j\neq i}}^{n} g_{ij} \qquad (12.23)$$

进而由定理12.1

$$y_{ij}\leq 0 \qquad (12.24)$$

所以由式（12.22）得到

$$y_{ij}=-g_{ij} \qquad (12.25)$$

现在由（12.23）和（12.25）可以看出：对所有i

$$y_{ii}\geq\sum_{\substack{j=1\\ j\neq i}}^{n}|y_{ij}| \qquad (12.26)$$

图 12.4 星树

一个实对称矩阵$Y=[y_{ij}]$是超优势的，如果它满足式（12.24）和（12.26），即

1. y_{ij}是非正的，对于所有的i和$j\neq i$；
2. $y_{ii}\geq\sum_{\substack{j=1\\ j\neq i}}^{n}|y_{ij}|$ 。

由（12.23）和（12.25）可以很容易地得到N的电导表达式：

$$g_{ij}=-y_{ij},\ j\neq i,\ i\neq 0,\ j\neq 0$$

$$g_{i0}=y_{ii}-\sum_{\substack{j=1\\ j\neq i}}^{n}|y_{ij}|,\quad i=0$$

如果矩阵Y满足式（12.24）和（12.26），那么上面所计算出的电导 g_{ij} 将是非负的。这样，我们就已经证明了下面的定理。

定理12.6 一个实对称矩阵Y可实现为（$n+1$）节点电阻 n 端口网络（具有星树端口图且不含有负电导）的短路电导矩阵，当且仅当Y可表为超优势形式。 □

12.2.3 线性树端口图

下面考虑一个具有线性树端口图T的（$n+1$）节点电阻n端口网络的Y矩阵。T的顶点依次地从末顶点开始标号，顶点 i 和$i+1$组成了端口 i 的正、负参考端点（图12.5）。由于T中所有端口是相同指向的，由定理12.1，

$$y_{ij}\geqslant 0,\ \text{对所有}\ i\ \text{和}\ j \tag{12.27}$$

而且由定理12.3，对所有 i，我们有

$$y_{ii}\geqslant y_{i,i+1}\geqslant\cdots\geqslant y_{in} \tag{12.28}$$

$$y_{ii}\geqslant y_{i-1,i}\geqslant\cdots\geqslant y_{1i} \tag{12.29}$$

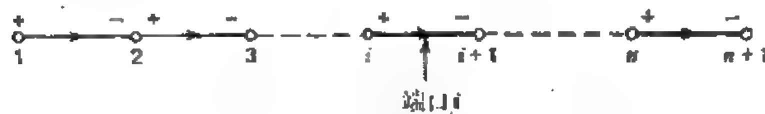


图 12.5 线性树

下面我们推导用Y矩阵中元素所表示的电导g_{ij}的简式。

假定我们将除端口$i-1$、i、$j-1$和 j 以外的所有端口都短路，这里$i>1$和$i+1<j\leqslant n$。那么所得到的网络如图12.6所示，图中仅对我们所感兴趣的电导作了标记。现在可容易看出

$$y_{i-1,i-1}=e+c$$

$$y_{i-1,j}=e$$

$$y_{i,j-1}=g_{ij}+c+d+e$$

$$y_{i,j}=e+d$$

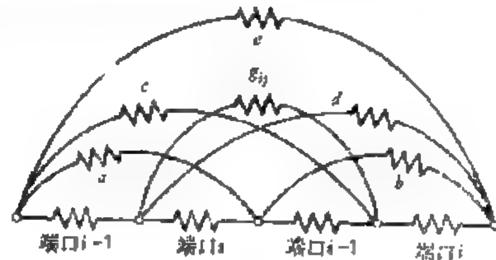


图 12.6

由上式得到

$$g_{ij}=(y_{i,j-1}+y_{i-1,j})-(y_{ij}+y_{i-1,j-1}),\quad i>1,\ i+1<j\leqslant n \tag{12.30}$$

我们还可以证明如下关系：

$$g_{i,i+1}=(y_{ii}+y_{i-1,i+1})-(y_{i,i+1}+y_{i-1,i}),\qquad 1<i<n$$

$$g_{1j}=y_{1,j-1}-y_{1j},\quad 2\leqslant j\leqslant n$$

$$g_{1,n+1}=y_{1n}$$

$$g_{i,n+1}=y_{in}-y_{i-1,n},\quad 1<i\leqslant n \tag{12.31}$$

一个实对称矩阵$Y=[y_{ij}]$是单调下降的，如果

1. $y_{ij} \geqslant 0$，对于所有i和j。

2. $y_{ij} \geqslant y_{i,j+1}$，对于$j \geqslant i$。

3. $y_{ij} \geqslant y_{i-1,j}$，对于$j \geqslant i$。

4. $y_{i,j-1} + y_{i-1,j} \geqslant y_{ij} + y_{i-1,j-1}$，对于$i>1$，$i+1 \leqslant j \leqslant n$。

由式（12.27）到（12.31），我们得到下面定理。

定理12.7 一个实对称矩阵$Y=[y_{ij}]$可实现为$(n+1)$节点电阻n端口网络（具有线性树端口图，且不含有负电导）的短路电导矩阵，当且仅当Y可表为单调下降形式。 □

已知一个单调下降矩阵Y，文献[12.3]给出了一个获得实现Y的n端口网络电导的步骤如下：

1. 首先由已知单调下降的$n \times n$阶矩阵Y构造一个新的$n \times n$矩阵Y'，以使得

a. Y'的第一行与Y的第一行相同；

b. 对所有$i>1$，Y'的第i行是Y的第i行和第（$i-1$）行的差。

2. 然后构造另一个$n \times n$矩阵Y''，以使得

a. Y''的最后一列与Y'的最后一列相同；

b. 对所有$i<n$，Y''的第i列是Y'的第i列和第（$i+1$）列的差。

因此可以看到：对所有$j>i$，$g_{ij}=y''_{i,j-1}$。

例如，对于单调下降矩阵

$$Y = \begin{bmatrix} 5 & 4 & 3 & 1 & 1 \\ 4 & 5 & 3 & 1 & 1 \\ 3 & 3 & 11 & 8 & 5 \\ 1 & 1 & 8 & 9 & 5 \\ 1 & 1 & 5 & 5 & 6 \end{bmatrix}$$

相应的Y'和Y''矩阵是

$$Y' = \begin{bmatrix} 5 & 4 & 3 & 1 & 1 \\ & 1 & 0 & 0 & 0 \\ & & 8 & 7 & 4 \\ & & & 1 & 0 \\ & & & & 1 \end{bmatrix}$$

$$Y'' = \begin{bmatrix} 1 & 1 & 2 & 0 & 1 \\ & 1 & 0 & 0 & 0 \\ & & 1 & 3 & 4 \\ & & & 1 & 0 \\ & & & & 1 \end{bmatrix}$$

$$
=\begin{bmatrix} g_{12} & g_{13} & g_{14} & g_{15} & g_{16} \\ & g_{23} & g_{24} & g_{25} & g_{26} \\ & & g_{34} & g_{35} & g_{36} \\ & & & g_{45} & g_{46} \\ & & & & g_{56} \end{bmatrix}
$$

在上面的计算过程中，我们没有给出矩阵Y'和Y''对角线下面的元素，这是因为在计算电导g_{ij}中，我们对这些元素不感兴趣。

12.2.4 端口变换

下面考虑两个由相同的电阻网络构成的、不同的$(n+1)$节点n端口网络N和N^*。设T和T^*分别是N和N^*的端口图。现在我们推导这些n端口网络的短路电导矩阵Y和Y^*之间关系表达式。

设Q_f和Q_f^*分别是N和N^*对应T和T^*的基本割集矩阵。那么

$$Y=Q_fG_eQ_f^t$$

和

$$Y^*=Q_f^*G_e(Q_f^*)^t$$

这里G_e是所讨论网络的对角边电导矩阵。

现在考虑图$T'=T\cup T^*$。我们可以写出对应于T的T'的基本割集矩阵为

$$\begin{matrix} T-T^* & T^* \\ [P & M] \end{matrix}$$

我们可将Q_f^*表示为

$$Q_f^*=M^{-1}Q_f \tag{12.32}$$

所以，

$$\begin{aligned} Y^* &=Q_f^*G_e(Q_f^*)^t \\ &=M^{-1}Q_fG_eQ_f^t(M^{-1})^t \\ &=M^{-1}Y(M^{-1})^t \end{aligned} \tag{12.33}$$

现在由式（11.8）则得

$$V_p^*=M^tV_p$$

这里V_p和V_p^*分别是N和N^*的端口电压向量。所以

$$V_p=(M^{-1})^tV_p^* \tag{12.34}$$

联立式（12.33）和（12.34），我们得到下面定理。

定理12.8 设N和N^*是两个由相同电阻网络构成的n端口网络。如果$V_p=KV_p^*$，那么

$$Y^*=K^tYK$$

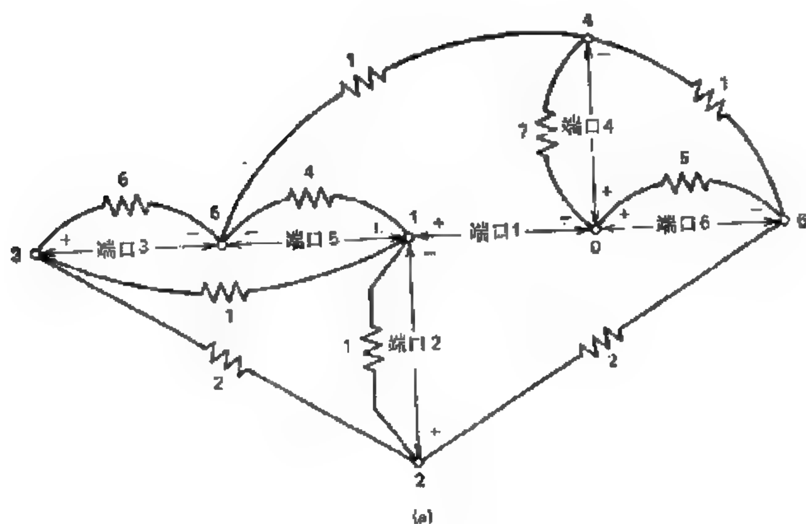
□

注意：上面定理中的矩阵K可以很容易地通过对T和T^*的观察而得到。例如，考虑图12.7所示的两个n端口网络。这两个网络都是由同一个母网络构造出来的。N的短路电导矩阵是

$$
Y=\begin{bmatrix}
3 & 2 & 0 & 1 & -1 & 2\\
2 & 5 & -2 & 0 & 2 & 2\\
0 & -2 & 9 & 0 & -3 & 0\\
1 & 0 & 0 & 9 & -1 & -1\\
-1 & 2 & -3 & -1 & 8 & 0\\
2 & 2 & 0 & -1 & 0 & 8
\end{bmatrix}
$$

矢量V_p和V_p^*的关系为

$$
V_p=\begin{bmatrix} v_1\\ v_2\\ v_3\\ v_4\\ v_5\\ v_6 \end{bmatrix}=\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0\\
-1 & 1 & 0 & 0 & 0 & 0\\
0 & 0 & 1 & 0 & -1 & 0\\
0 & 0 & 0 & -1 & 0 & 0\\
1 & 0 & 0 & 0 & -1 & 0\\
0 & 0 & 0 & 0 & 0 & -1
\end{bmatrix}\begin{bmatrix} v_1^*\\ v_2^*\\ v_3^*\\ v_4^*\\ v_5^*\\ v_6^* \end{bmatrix}=KV_p^*
$$



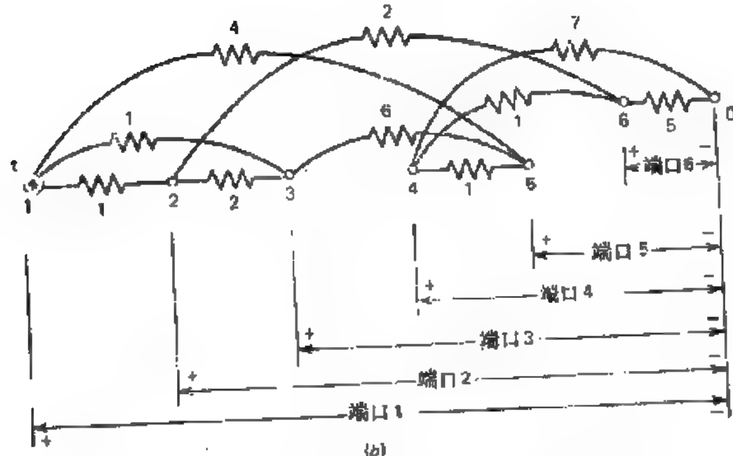


图 12.7 (a)N; (b)N^*(所有电导单位都是西门子)。

现在我们可以验证$Y^*=K^tYK$：

$$Y^*=K^tYK=\begin{bmatrix} 6 & -1 & -1 & 0 & -4 & 0 \\ -1 & 5 & -2 & 0 & 0 & -2 \\ -1 & -2 & 9 & 0 & -6 & 0 \\ 0 & 0 & 0 & 9 & -1 & -1 \\ -4 & 0 & -6 & -1 & 11 & 0 \\ 0 & -2 & 0 & -1 & 0 & 8 \end{bmatrix}$$

12.3 $(n+1)$节点电阻n端口网络实现——I

本节，我们考虑如何实现一个实对称$n\times n$矩阵Y作为不含负电导的$(n+1)$节点电阻n端口网络的短路电导矩阵。对这个问题，Cederbaum和Guillemin提出两种基本方法。

Cederbaum的方法（文献[12.1]）需要把矩阵Y分解成为矩阵连乘积$Y=Q_fG_eQ_f^t$，其中Q_f的元素为-1、1、0，G_e是一个对角矩阵，它的对角线上的非负元素代表所求的n端口网络N的边导纳。如果$[U\ \ Q_f]$能够表示为一个连通图的基本割集矩阵，那么Y的实现即可完成。这个连通图就是增广n端口网络$\hat{N}$的图，而Q_f的列对应于表示所求网络N的电导的边。事实上，正如我们在本节的后面看到的那样，Cederbaum方法也可适用于零度为n的n端口网络的开路电阻矩阵的实现。

Guillemin的方法（文献[12.3]）需要确定端口图（如果存在），并把矩阵Y变换为对应于星树端口图的矩阵Y'。如果Y'是超优势的话，那么Y的实现则通过观察即可完成（正如我们在上一节所看到的那样）。通过对端口的适当的重新定义，同样的网络也可实现给定的Y。

现在我们描述将Y分解成三乘积$Y=Q_fG_eQ_f^t$的Cederbaum算法。下节我们将考虑作为一个基本割集矩阵$[U\ \ Q_f]$的实现。以Guillemin方法为基础来实现$(n+1)$节点n端口网络Y矩阵的算法，将在12.5节讨论。

设$\hat{q}_1$、$\hat{q}_2$、…表示矩阵Q_f的列矢量，g_1、g_2、…表示G_e的对角线元素。$\hat{q}_i$为Q_f的第i列，而g_i是G_e的第i个对角元素。注意：$\hat{q}_i$对应于g_i，进而$\hat{q}_i$中的元素表示为q_{1i}、q_{2i}、…、q_{ni}。

Cederbaum算法从矩阵$Y^{(0)}=Y$开始来构造一系列矩阵$Y^{(0)}$、$Y^{(1)}$、…，其中每一步对应于从网络中移去一个适当标志的电导，这个过程直到获得了一个对角矩阵，或者看出这个分解是不可能时才停止。

在算法的每一步中，我们需要考虑两种情况。利用矩阵Y，讨论每种情况下应采取的步骤。

情况1　Y是具有k个零对角元素的对角矩阵。令Y'是划出Y中含有零对角元素的所有行和列后所得到的矩阵。我们很容易看出，这种情况下的分解是

$$Y=\begin{bmatrix} U \\ 0 \end{bmatrix} Y' \begin{bmatrix} U \\ 0 \end{bmatrix}^t$$

这里假定Y的零对角元素出现在最后k个位置上。例如，矩阵

$$\begin{bmatrix} 3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

可以分解为

$$Y = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \hline 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 3 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix} \left[\begin{array}{ccc|cc} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{array}\right]$$

情况 2　在对角线外，Y有些非零元素。

在这种情况下，我们首先选择一个Y的对角线以外的最小非零元素，我们假定y_{12}是Y的这样一个元素。如果Y是可实现的，那么由推论12.3.1可知：在所求的网络N中，一定存在一条边，其电导值为$|y_{12}|$。换言之，G_e必定有一个对角元素等于$|y_{12}|$。不失一般性，我们假定

$$g_1 = |y_{12}|$$

我们能够确定对应于g_1的列矢量$\hat{q}_1$步骤如下：

很显然，电导值为$|y_{12}|$的边生成了N的两个端口1和2。因此，$\hat{q}_1$的前两个元素是非零的。由推论12.3.1可知，Q_f的前两行在除第一列以外的任何列上不可能同时有非零元素。所以我们任意给定$\hat{q}_1$的第一个元素为$+1$，因此固定了具有电导g_1的边的指向，与端口1的指向相关联。然后由定理12.1可知，$\hat{q}_1$中第二个元素的符号与y_{12}的符号相同。这样

$$\hat{q}_1 = \begin{bmatrix} 1 \\ \pm 1 \\ \vdots \\ \vdots \end{bmatrix}$$

注意：如果y_{rs}是对角线两旁的具有最小值的非零元素，那么$\hat{q}_1$的第r和第s个元素即可用同样的方法确定。

剩下的问题就是确定$\hat{q}_1$的其它元素。这个问题可由如下的规则来解决，这个规则是建立在定理12.2基础上的：

$$q_{j1} = \begin{cases} 1, & \text{如果} y_{12}\ y_{1j}\ y_{2j} > 0 \text{ 且} y_{1j} > 0 \\ -1, & \text{如果} y_{12}\ y_{1j}\ y_{2j} > 0 \text{ 且} y_{1j} < 0 \\ 0, & \text{其它} \end{cases}$$

这样我们就完成了两种情况的讨论。

下一步，我们必须找出Q_f的第二列和对应的G_e的对角线元素g_2。考虑网络$N^{(1)}$的短路电导矩阵$Y^{(1)}$即可完成这一步，这里$N^{(1)}$是从N中移去电导g_1后得到的网络，即令N中$g_1=0$。

我们除掉Q_f的第一列以及G_e的第一行和第一列，并设$Q_f^{(1)}$和$G_e^{(1)}$表示除掉后所得到的矩阵。那么可以看出

$$
\begin{aligned}
Y^{(1)} &= Q_f^{(1)} G_e^{(1)} (Q_f^{(1)})^t \\
&= Y - \hat{q}_1 [g_1] (\hat{q}_1)^t
\end{aligned}
\qquad (12.35)
$$

因此

$$y_{ij}^{(1)} = y_{ij} - q_{i1} q_{j1} g_1 \qquad (12.36)$$

注意：如果矩阵是可实现的，那么Y的所有零元素在$Y^{(1)}$中仍然是不变的，并且非零元素y_{ij}也不改变其符号或增加其量值。因此，在这种情况下，矩阵$Y^{(1)}$将比Y具有更多的对角线以外的零元素。

假设$Y^{(1)}$是对角矩阵，那么情况1下所描述的方法，将给出Q_f的所有剩余列和G_e的剩余对角线元素。

如果$Y^{(1)}$不是对角矩阵，那么就重复情况2下所描述的过程，以获得电导g_2和Q_f的第二列$\hat{q}_2$为止。矩阵$Y^{(2)}$由下式给出

$$Y^{(2)} = Y^{(1)} - \hat{q}_2 [g_2] (\hat{q}_2)^t$$

如果Y是可实现的，那么显然序列$Y^{(0)}$、$Y^{(1)}$、$Y^{(2)}$、…中的每个矩阵都将落入前面所描述的两种情况之一，因此每个矩阵是可分解的。

如果在任意一步，相应矩阵对角线以外的元素的值开始增加或者改变符号，那么分解就是不可能的，因此在这一步就可以终止算法。在有些情况下分解是可能的，但Q_f不是单位模矩阵或者G_e具有负的对角线元素，或者两者皆有。这时，Y也是不可能实现的。

我们用下面的矩阵来说明Cederbaum算法。

$$
Y = Y^{(0)} = \begin{bmatrix}
3 & 2 & 1^* & -2 & 0 \\
2 & 5 & -2 & -2 & 1 \\
1 & -2 & 3 & 0 & 0 \\
2 & -2 & 0 & 5 & 3 \\
0 & 1 & 0 & 3 & 4
\end{bmatrix}
$$

因为Y不是对角矩阵，所以情况1不会出现。因此我们按情况2进行。

y_{13}是Y矩阵中最小值非零元素，所以我们令

$$g_1 = |y_{13}| = 1$$

$$q_{11} = 1$$

$$q_{31} = 1 \quad （因为y_{13}是正的）$$

然后我们利用下面的规则确定Q_f的第一列$\hat{q}_1$中剩余的元素：

$$
q_{j1}=\begin{cases}1, & \text{如果}\ y_{1j}y_{3j}y_{13}>0\ \text{且}\ y_{1j}>0\\ -1, & \text{如果}\ y_{1j}y_{3j}y_{13}>0\ \text{且}\ y_{1j}<0\\ 0, & \text{其它}\end{cases}
$$

例如，$y_{12}y_{32}y_{13}<0$，则$q_{21}=0$，这样，我们得到

$$
\hat{q}_1=\begin{bmatrix}1\\0\\1\\0\\0\end{bmatrix}
$$

然后得到$Y^{(1)}$：

$$
Y^{(1)}=Y^{(0)}-\hat{q}_1[g_1](\hat{q}_1)^t
$$

$$
=\begin{bmatrix}2 & 2 & 0 & -2 & 0\\ 2 & 5 & -2 & -2 & 1^*\\ 0 & -2 & 2 & 0 & 0\\ -2 & -2 & 0 & 5 & 3\\ 0 & 1 & 0 & 3 & 4\end{bmatrix}
$$

在$Y^{(1)}$中，（2，5）元素具有最小非零值。所以得到

$$
g_2=1
$$

$$
q_{22}=1
$$

$$
q_{52}=1
$$

同样可得到$\hat{q}_2$剩余的元素，因此

$$
\hat{q}_2=\begin{bmatrix}0\\1\\0\\0\\1\end{bmatrix}
$$

则矩阵$Y^{(2)}$为

$$
Y^{(2)}=Y^{(1)}-\hat{q}_2[g_2](\hat{q}_2)^t
$$

$$
=\begin{bmatrix}2 & 2 & 0 & -2^* & 0\\ 2 & 4 & -2 & -2 & 0\\ 0 & -2 & 2 & 0 & 0\\ -2 & -2 & 0 & 5 & 3\\ 0 & 0 & 0 & 3 & 3\end{bmatrix}
$$

以这样的方法继续下去，我们得到：

$$g_3=2,\qquad \hat{q}_3=\begin{bmatrix}1\\1\\0\\-1\\0\end{bmatrix}$$

以及

$$Y^{(3)}=\begin{bmatrix}0&0&0&0&0\\0&2&-2^*&0&0\\0&-2&2&0&0\\0&0&0&3&3\\0&0&0&3&3\end{bmatrix},\quad g_4=2,\quad \hat{q}_4=\begin{bmatrix}0\\1\\-1\\0\\0\end{bmatrix}$$

$$Y^{(4)}=\begin{bmatrix}0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&3&3^*\\0&0&0&3&3\end{bmatrix},\quad g_5=3,\quad \hat{q}_5=\begin{bmatrix}0\\0\\0\\1\\1\end{bmatrix}$$

上面打星号的是每一步中选择要考虑的元素。下面是$Y^{(0)}$的一个分解：

$$Y^{(0)}=\begin{bmatrix}3&2&1&-2&0\\2&5&-2&-2&1\\1&-2&3&0&0\\-2&-2&0&5&3\\0&1&0&3&4\end{bmatrix}=\begin{bmatrix}1&0&1&0&0\\0&1&1&1&0\\1&0&0&-1&0\\0&0&-1&0&1\\0&1&0&0&1\end{bmatrix}$$

$$\times\begin{bmatrix}1&0&0&0&0\\0&1&0&0&0\\0&0&2&0&0\\0&0&0&2&0\\0&0&0&0&3\end{bmatrix}\begin{bmatrix}1&0&1&0&0\\0&1&0&0&1\\1&1&0&-1&0\\0&1&-1&0&0\\0&0&0&1&1\end{bmatrix}$$

我们应指出，上面所讨论的Cederbaum的分解算法对任何最高的矩阵都是适用的：即如果一个最高的矩阵K能够分解成为PDP^t，这里P是单位模矩阵，而D是正对角矩阵，那么用这个算法就可找出这样的分解形式。因此我们可以看出，这个算法也可用来分解一个零度为n的n端口网络的开路电阻矩阵，使其成为三乘积$B_fRB_f^t$的形式。而且，这种实现（$n+1$）节点n端口网络的Y矩阵的Cederbaum算法，对于实现零度为n的n端口网络的Z矩阵也是有效的。

在确定了B_f或Q_f之后，Cederbaum算法的下一步就是确定对应于这些矩阵的有向图。假设我们已经确定了对应于B_f或Q_f的无向图。那么可以很容易通过对B_f或Q_f的观察来确定适当的有向图（如果存在的话），这种情况可能会出现在习题（6.1）中。

这样，就要提出一个算法能够实现一个给定无向图的回路和割集矩阵。这样的算法将在下节讨论。

12.4 割集和回路矩阵的实现

本节我们将讨论构造一个具有给定回路或割集集合的无向图问题。不失一般性，我们可假定：已知所要构造图的基本割集矩阵Q_f或基本回路矩阵B_f。

设B_f是关于生成树T的基本回路矩阵，那么可以将B_f写成$B_f=[U\ \ F]$，这里F、U分别对应于T和生成补树。由B_f的定义可知，F的每一行中的非零元素对应于形成T中适当的连支端点之间的一条路径的支路，换言之，F的每一行对应于T中的一条路径。因此，F称为关于T的树-路径矩阵。

已知矩阵F，由式（6.13）可知，$Q_f=[F'\ \ U]$。所以我们说，B_f和Q_f的实现不是两个无关的问题。它们都等效于树-路径矩阵F的实现。

实现F的主要问题就是要确定具有对应于F中列的支路的树结构，以使F中每行所描述的树的路径都在这个结构中。很显然，从这样确定的树中构造出所需的图，仅仅需要对树添加一些F中每行的适当顶点之间的连支就行了。

现在我们讨论文献[12.4]提出的实现无向图G的树-路径矩阵F的一个算法。

12.4.1 前　奏

我们首先考虑一些规则，在讨论F的实现之前，我们可以利用这些规则来化简F的复杂度。

如果已知F矩阵是一个直接和矩阵，即F可划分如下：

$$\begin{bmatrix} F_1 & 0 & \cdots & 0 \\ 0 & F_2 & \cdots & 0 \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ 0 & 0 & \cdots & F_n \end{bmatrix}$$

那么每个子图F_1、F_2、…、F_n都可以分别实现，并且已实现的子图（树）可以被任意地相互连接，以得到所需的树T。

如果F中存在一组相同行，则说明图G中存在一组并行的连支。为了实现关联树T，可以在F中只留下这一组相同行中的一行，而删去所有其他的相同行。只有一个非零元素的行也可以被压缩，因为它对T中树支的相互连接不产生制约。

如果出现一个仅存在于指定的n条路径中的一条特定树支，则从F中划去关联的列后就可实现树。这个支路可以加到适当路径的两个端点之一。进而，如果F中的某k个列是相同的，则划去$k-1$个这样的列后就足以实现所得的矩阵，然后对应于保留下来列的树支，再用一个任意方式排序的树支的串联-连接集来置换。

利用上述规则，通过连续地划去多余的行和列，就可对一个给定的矩阵F在复杂度上进行化简。这个程序叫做实现F的前奏。

12.4.2 末梢树支的识别

实现F矩阵的上述算法的第一个主要步骤，就是实现F的树末梢树支的识别。现在我们讨论识别这样的末梢树支的一个方法。这个方法对任何一般的F矩阵都是适用的，而且如果F能有几种不同实现的话，那么至少存在一种以被识别树支作为末梢树支的实现。

设$W=[w_{ij}]$是一个对称矩阵，且定义为

$$W=F^tF \tag{12.37}$$

这里乘法和加法运算都是在实数域进行的。显然，元素w_{ij}等于由 F 指定的含有树支 i 和 j 的路径的数目。注意，我们已经假定，W的第 i 行或列对应于树T的第 i 条树支。

设G为实现给定矩阵F的图，T为相应的生成树。如果我们将G中的每条边视为1S的电导，并且$Y=[y_{ij}]$是所需的n端口网络对应于端口图T的短路导纳矩阵，那么我们可以看出，对于$i\neq j$，有$w_{ij}=|y_{ij}|$。因此，我们得到如下的性质，这些性质仅仅是定理12.3、12.4和12.5的重现。

性质12.1 如果树支 i、j 和 k，在T中其他树支都短路时，按如上次序形成了一个线性树的话，那么$w_{ij}\geq w_{ik}$。

性质12.2 设W中第 i 行元素的最大值为M_i，而这个元素 M_i同时只出现在这一行的第 j、k、l、…列上，那么对应于 i、j、k、l、…的树支形成了T的一个子树。

性质12.3 设W中第 i 列的元素最小值为m_i（m_i 可能为零），而这一行的最小元素同时只出现在第p、q、r、…列上，那么T中除p、q、r、…以外的树支形成了T的一个子树。

为了进一步讨论，我们需要引入如下的定义。

L集合是构成T的一个子树的树支集合，在这个子树中，仅仅存在一个顶点与其在T中的补所共有，并且仅有集合中的一条树支（称之为L集合的引导树支）与那个顶点相关联。

为了说明上述定义，我们观察图12.8，图中的l、k、m和n构成了一个以l作为引导树支的L集合，而树支k、m、和n就不能构成一个L集合。一个树的末梢树支可以看成是L集合的最简形式。

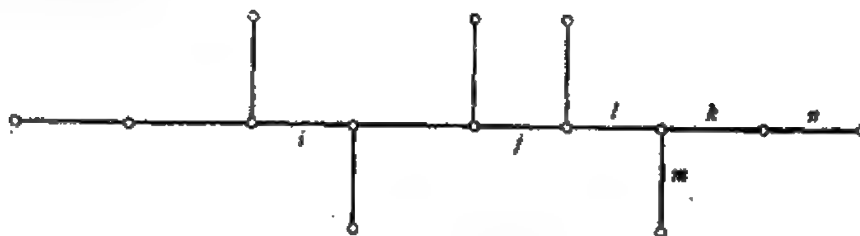


图 12.8 L集合的解释

很容易看出，T中除L集合之外的剩余部分也是T的一个子树。对任意不是T的末梢树支的树支r，总存在两个L集合，每个集合都以r作为引导树支。

现在我们考察T的一个子树的末梢树支也可是T的一个末梢树支的这种情况。设T中

树支可以划分为两个集合 S 和 R，使得 R 中的树支形成 T 的一个子树。进而假定下面的关系成立：

$$w_{r_i s_i} = w_{r_i s_j},\ 对于所有 r_i \in R 和 s_i,\ s_j \in S \qquad (12.38)$$

下面我们证明，通过短路 R 中所有树支而获得的新的一个末梢树支，或者是 T 的一个末梢树支，或者是 T' 的一个末梢树支，这里 T' 是由对应于给定图的一个2-同构图中 T 的树支的边所构成的。

如果关系

$$w_{rs} = 0,\ 对每一个 r \in R 和 s \in S \qquad (12.39)$$

满足的话，那么矩阵 F 就是一个直接和矩阵，而且对应于含在 S 和 R 中树支的 F 的子矩阵也可分开实现。在这种情况下，由 S 中树支所构成子树中的一个末梢树支，显然在整个实现中可以构成 T 的一个末梢树支。

另一方面，假设存在某个 $r_i \in R$，对于这个 r_i，$w_{r_i s}(s \in S)$ 不等于零。因为 R 中的树支形成了 T 的一个子树，所以集合 S 可以划分成一些分离的 L 集合。我们首先证明每个 L 集合都是 T 的一个线性子树。

设在以 s_i 作为引导树支的一个 L 集合中，存在着树支 s_j 和 s_k，并使得 s_i、s_j 和 s_k 在 T 的其他树支短路时形成一个星树。因为使 s_k 和 r_i 都出现的路径与那些使 s_j 和 r_i 都出现的路径是不相同的，而且 s_i 存在于两种形式的路径中，因此我们得到

$$w_{r_i s_i} \geqslant w_{r_i s_j} + w_{r_i s_k} \qquad (12.40)$$

但是由(12.38)可知，$w_{r_i s_i} = w_{r_i s_j} = w_{r_i s_k}$，而且它们都不为零。所以，式(12.40)不可能成立。因此，s_i、s_j 和 s_k 应该包含在 T 的一个线性子树中。推广这个结论可以看出，在每一个将 S 划分的 L 集合中的所有树支，都构成了 T 的一个线性子树。

这样，当 R 中树支被短路时，就可以由 G 得到一个新图 G_1，在一般情况下，G_1 可以分成一些不可分片，其中每个不可分片都含有一个或更多个将从 S 划分的 L 集合。

设 G_1 的一个特定的不可分片仅仅含有一个 L 集合，它的树支构成了一个以 x 和 y 作为末梢树支的线性子树 L_1。考虑 F 关于含有 L_1 中树支的路径的特性。这些路径中的某些路径可以仅含有来自 L_1 中的树支，其他的路径则既含有 L_1 中的树支，也含有些不在 L_1 中的树支，但是后者中的每一路径都含有 L_1 中的所有树支，这是因为有关 W 的元素都是相等的。因此，由 L_1 中树支和那些其路径仅含有 L_1 中树支的连支所组成的 G_1 的子图，能够"旋转"得到一个与 G_1 2-同构的图。如果在一个图中，末梢树支是 x，那么在另一个图中，末梢树支就是 y。

例如，对于图12.9(a)所示的图 G，相应的图 G_1 含有三个不可分片，如图12.9(b)所示。我们注意到，s_6 是图 G_1 生成树的一个末梢树支，图12.9(c)所示图 G_1' 是 G_1 的2-同构图，并且 s_4 是 G_1' 对应树的一个末梢树支。这样就存在着一个以 s_4 作为末梢树支的 G 的树-路径矩阵 F 的另一个实现 G'（图12.9(d)）。利用我们后面将要介绍的算法，就可以识别出实现 F 矩阵的某个树的末梢树支。运用这个算法，树支 s_1、s_4、s_5、s_7、s_9 和 s_{12} 中的任意一个都可以成为实现 F_1 矩阵的树的一个末梢树支。对于其中的每种选择，都存在以特定

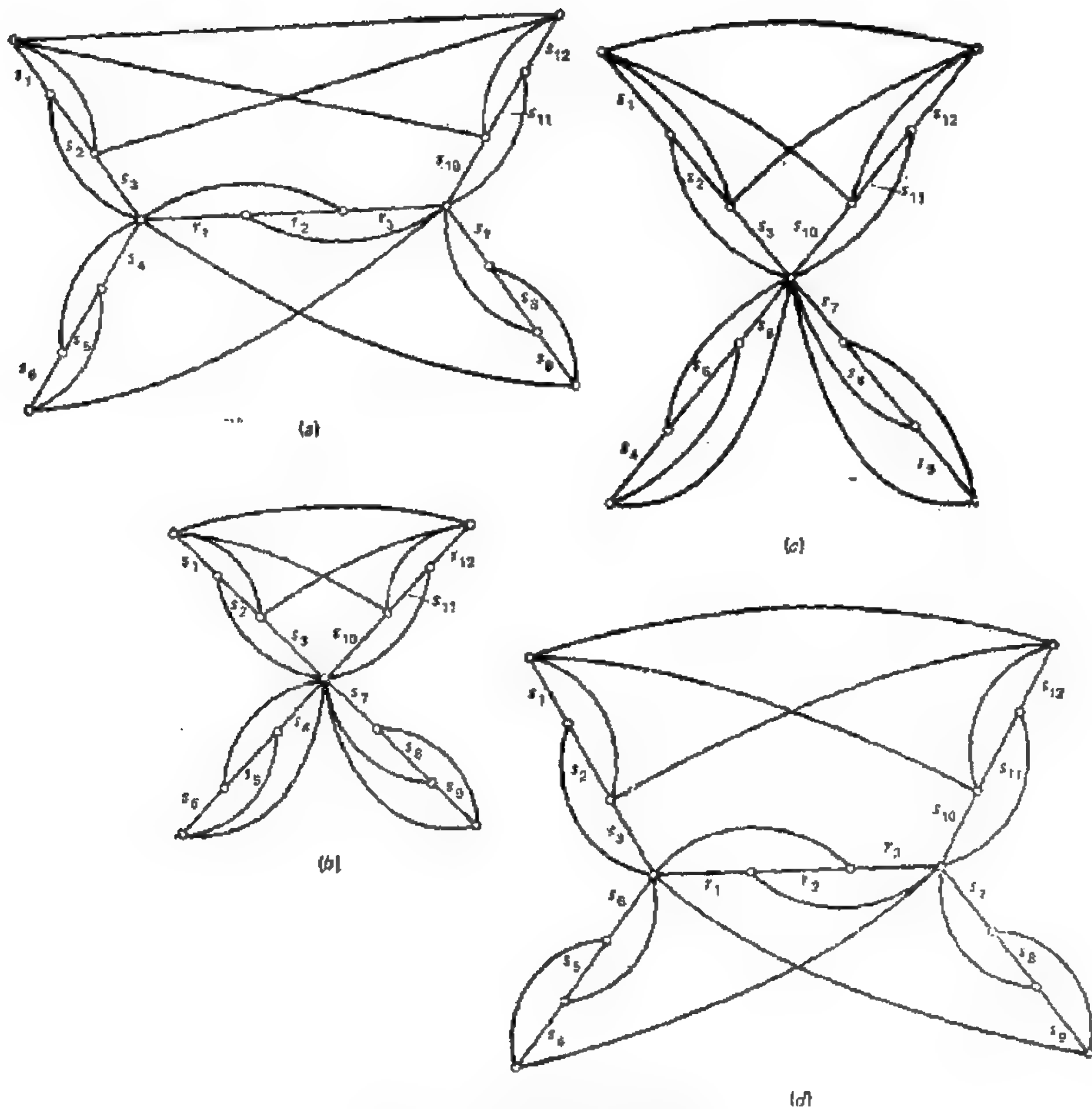


图 12.9 由导出图识别末梢树支的例子
(a)G; (b)G_{11}; (c)G'_{11}; (d)G'。

树支作为末梢树支的F的树实现。

上面的讨论产生了下面一个问题：

当T中树支划分成为具有式(12.38)性质的两个集合R和S以后，那么为了得到T的一个确实存在于集合S中的末梢树枝，我们就得考虑短路R中树支之后所得的图，以及寻找一个由S中树支所构成的树的末梢树支。这样确定的末梢树支就可认为是所求图的树的一个末梢树支。

寻找实现给定F矩阵的树T中一个末梢树支的算法如下：

第一步　获得矩阵$W = F^{t}F$。

第二步 考虑W的任意i行。令m_i（可能为零）是这一行的最小元素。如果在第i行中仅存在一个具有最小值的元素w_{ij}，那么j就是一个末梢树支。否则，设第i行中等于m_i的元素所在列组成了集合$S=\{s_1,s_2,\cdots,s_n\}$。很显然，集合S包含了T的一个末梢树支。设不在S中的T的树支子集表示为集合R。

第三步 考虑任意行$r_i\in R$。如果$w_{r_is_i}>w_{r_is_j}$，这里s_i，$s_j\in S$，那么就从S中划去s_i。把这一判据应用到第i行中所有相关元素对，并从S中划去所有可能的元素，再将简化后的集合表示为S_1，而它的补集为R_1。

第四步 对W中其他所有行重复第三步，每次选择一行r_j，并使得r_j是R集合中的一个元素，且这时的R集合表示为R_i。这一过程直到出现下面两种情况之一为止：

a. S_k仅含有一个元素，那么此元素便对应于末梢树支；

b. S_k不可再简化了，即对所有$r_i\in R$和集合S_k中的每一对s_i和s_j，$w_{r_is_i}=w_{r_is_j}$。

第五步 如果集合S_k中的元素个数大于1（第四步中的情况b），那么任意选择一个S_k中的元素放在第三步中的r_i的位置上，然后继续做第三、四、五步，继续对S_k进行化简，即消去S_k中的任意一个元素，并将其添加到R_i中去。重复这些步骤，最终可导出一个仅含有一个元素的集合S_p，这个元素就可看作是T的一个末梢树支。

下面给出上述算法的证明。我们首先注意到，由性质12.3可知，在第二步中的集合R的元素构成了一个子树。所以，补集S必须至少含有一个末梢树支。进而我们还注意到，如果S_j是S中的一个元素，那么由性质12.1可知，不含有i，而且以S_i作为引导树支的特定L集合中的所有元素都是在S中。

在第三步中，我们通过划去象s_k这样的元素，从而得到了一个简化集S_1，并使得$w_{r_is_k}>w_{r_is_j}$，$r_i\in R$且s_k，$s_j\in S$。我们继续注意到，S_1至少有一个T的末梢树支。这一事实是很明显的，（1）当s_k和s_j在两个不同于如上讨论的L集合中，或者（2）当s_k和s_j在同一个L集合中，树支r_i、s_k和s_j包含在T的一个线性子树中。另一方面，如果s_k和s_j在同一L集合中，以及树支r_i、s_k和s_j当T中所有其他树支都短路时形成了一个星树，那么这个L集合至少含有两个末梢树支。因此，在这种情况下，即使s_k是一个末梢树支，简化集S_1（删去s_k所得到的）也含有T的一个末梢树支。我们还附带地得到，当s_k被划去之后（性质12.1），出现在以r_i和s_k作为其端树支的树路径中的每条树支也可被划去。这样，再次完成第三步之后，包含在R中的树支集合就构成了T的一个子树。

在第四步中，通过选取W中的另一个适当的行，重复上面的循环过程，在每一阶段，简化集合s_i仅含有一个末梢树支，R_i中的树支继续形成T的一个子树。

在许多情况下，可以通过执行这一过程（第四步）来识别一个末梢树支，直到获得仅含有一个元素的S_k。但是，当不出现这种情况，即不可能用第四步进一步减少S_k中元素的数目时，这就意味着，对于所有的$r_i\in R_k$和所有的s_i，$s_j\in S_k$，$w_{r_is_i}=w_{r_is_j}$。这样，在这一阶段中，R_k的元素就形成了T的一个子树，而且R_k和S_k中的元素满足式（12.28）。

如果G_1是从G通过短路R_k中的支路所得到的图，那么如前所述，G_1就是一个可分图。其中G_{11}、G_{12}、…、G_{1q}为它的不可分片。集合S_k现在包含了G_1的一个生成树的树支。

在第五步中，我们任意选择S_k中的一个元素，比如G_{1i}中的一个支路，并将这个元素转移到集合R_k中，然后对于相应于这个元素的行执行第三步和第四步。很容易看出，这些步骤结束时，集合S_k中剩下的元素都是G_1的除G_{1i}之外所有不可分片中的树支。重复这些步骤，最终便导出一个含有仅出现在G_1的一个片中的树支的集合，因此也就导出了所求T本身的一个末梢树支。

以上的讨论证明了识别实现给定F矩阵的树的一个末梢树支算法的正确性。

12.4.3 增广路径

设b为如上所确定的一个末梢树支，F_1为通过划去F中对应于b的列后所得到的矩阵。如果得到了实现F_1的一个树T_1，那么仅当T中含有b的各路径具有一个公共末梢顶点的话，F的树T就可以通过把末梢树支加到T_1中来增长。但是，事情并不总是这样。例如，考虑下面给定的F矩阵：

$$
F=\begin{array}{c} \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \end{array}
\begin{array}{c}
\begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\
\left[\begin{array}{ccccc}
1 & 1 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 \\
0 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 1
\end{array}\right]
\end{array}
$$

我们可很容易识别支路1是一个末梢树支。然后，从F中划去第1列后所得到的矩阵F_1是

$$
F_1=\left[\begin{array}{cccc}
1 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 \\
1 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 \\
0 & 0 & 1 & 1
\end{array}\right]
$$

可以看出，F_1是通过图12.10(a)所示的T_1来实现的，但是，这个树不能够增长到现实F的树，这是因为对应于连支6和7的路径是不可能配合的。但是，这并不意谓着F是不可实现的，因为F_1的一种变换实现可能导致F的实现。事实上，图12.10(b)所示的树T'_1也可

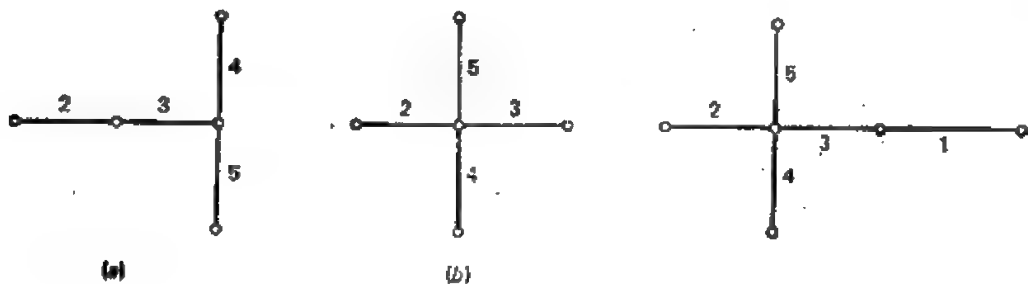


图 12.10 (a)树T_1；(b)树T'_1；(c)树T。

实现F_1，并且能够增长到实现矩阵F的树T（图12.10(c)）。

为了克服上面的困难，我们指出T_1中要添加路径，即通过附加的行来增广F_1，并使得实现增广矩阵的树恰好是所需要实现T中的末梢树支b的补。

设含有由F所规定的末梢树支的路径用p_1'、p_2'、…、p_r'来表示。假定所有这些路径都是可区别的，并且每条路径的支路数大于1。为了便于讨论，我们定义一个路径是平凡的，如果它存在着另一条与之相同的路径，或者它仅含有一条支路。这样，在上述假定下，p_1'、p_2'、…、p_r'中不存在平凡路径。这些路径中的末梢树支的补也是T中的一条路径。设这些路径为p_1、p_2、…、p_r。注意，所有这些路径在T中有一个公共的端顶点。

进而，任何实现F_1的树也都含有这些路径。现在我们描述一个用来增广F_1的方法，使得在实现F_1的每个树中，路径p_1、p_2、…、p_r有一个公共的端顶点。由这样的树，我们可以很容易地通过在这些路径的公共端顶点上加入树支b来生长出所需的树T。

考虑集合p_1、p_2、…p_r中的任意三条路径p_i、p_j和p_k。如果$p_i \oplus p_j = p_k$，则不存在路径p_i、p_j和p_k具有一个公共端顶点的现实。这点可由下面的事实看出，即p_i，p_j和$p_i \oplus p_j$仅在图12.11所示两种特定情况下才能都是一个树中的路径。这样如果p_1、p_2、…、p_r中的任何三条路径的环和等于空集的话，那么矩阵F就不可能实现了。为了检验这种情形的存在，我们需要分别考察C_r^3个路径集合。但是，由于我们的目的是实现F，所以我们仅考察上面的$r-2$个集合的一致性，然后再由$2r-3$个附加行增广F_1来保证，如果F是不可实现的话，那么增广后的F_1也是不可实现的。

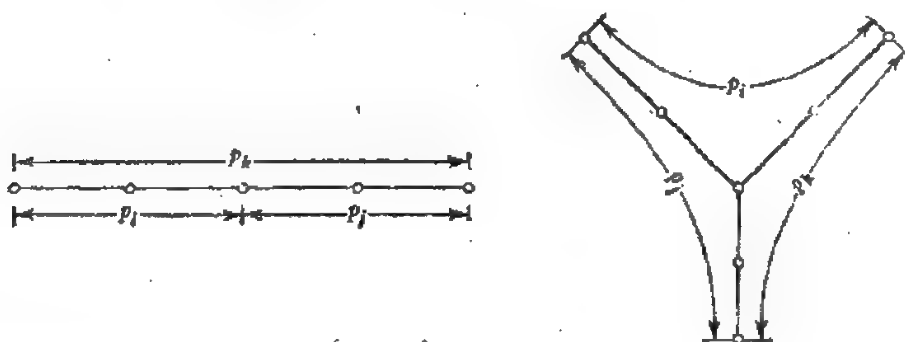


图 12.11

我们首先保证$p_1 \oplus p_2 \neq p_k$，$k=3$、4、…、r。如果对于某个k，$p_1 \oplus p_2 = p_k$，那么矩阵F当然是不可能实现的。完成这一检验之后，矩阵F就可由对应于如下路径的行来增广：

1．$p_1 \oplus p_2$，$p_1 \oplus p_3$，…，$p_1 \oplus p_r$。

2．$p_2 \oplus p_3$，$p_2 \oplus p_4$，…，$p_2 \oplus p_r$。

1中所指定的附加路径强加了这样一个条件，即路径p_2、p_3、…、p_r与p_1共享一个在实现增广F_1矩阵树中的公共端顶点，这是因为仅当p_1和p_k具有一个公共端顶点时，$p_1 \oplus p_k$是一条路径。同样地，第二个路径集合的存在的规定迫使p_3、p_4、…、p_r中的每条路径也和p_2同享一个公共的端顶点。由于T_1中不存在回路，并且预先的检验保证$p_1 \oplus p_2 \neq$

p_k（$k=3$、4、…，r），而且排除了T_1中存在如图12.11所示的那些路径，因此，p_1和p_2的公共端顶点也是每条路径p_k（$k=3$、4、…、r）的一个端顶点。这样，如果增广后的F_1是可实现的话，那么在这个矩阵任何实现的T_1中，路径p_1、p_2、…p_r具有一个公共的端顶点。在这种情况下，T_1可以生长成为实现F的树。另一方面，如果增广后的F_1矩阵是不可能实现的，那么这就意味着如下两点之一：

1.F_1是不可能实现的，或

2.不存在F_1的实现，在这个实现中p_1、p_2、…、p_r具有一个公共的端顶点。

在这两种情况下，F也是不可能实现的。

确定一个实现给定矩阵F的树T的步骤将在下面列出。

12.4.4 树的构造

我们首先对F反复应用前奏中的操作，直到获得一个不能再进行化简的矩阵F'为止。当得到一个实现F'的树T'后，实现F的树就可以由此而生成了。

下面我们先识别T'的一个末梢树支，再进行路径一致性的预先检验。从F'中划去对应于已产生末梢树支的列，然后再由前面所述的附加行来增广所得到的矩阵。对于这个增广矩阵应用关于行的前奏中的操作，其所得矩阵记为F'_1。

现在我们识别一个将要实现F'_1的树的末梢树支，并且继续上面相同的思路来寻找求出F'_2的过程。重复这一过程直到获得一个仅含有两列的矩阵F'_k为止。若F'不可能实现，亦即F不可能实现的情况下，这种情形在化简过程的某一步就将被识别出来。

如果F'是可实现的，那么从实现F'_k的树T_k开始，将末梢树支按它们被移去时相反的次序加上。前奏中操作的应用保证了两个含有末梢树支的路径的存在，因此，末梢树支的位置在每一步中也就唯一地固定了。这样，我们就能够构造出T'，因而构造出T，将对应于F中行的连支加到T上之后，图G就被实现了。

我们可以说，通过前奏中操作的应用将给定的F矩阵化简成为矩阵F'并不是必需的。这一化简过程可以直接应用到F本身，只要每一步所考虑的路径p'_1、p'_2、…、p'_r都是不同的，而且每个路径都含有数目大于1的树支。如果在T_k生长T的过程中，在任意一步中不可能得到含有末梢树支的两个不同路径，那么末梢树支就可加到有关路径的任意一端。

下面我们用两个例子来说明本节的讨论。

首先，设需要实现的矩阵为

$$F=\begin{matrix} 1 & 2 & 3 & 4 \\ \end{matrix}$$
$$F=\begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

我们可以看出，给定的矩阵不可能通过前奏中的操作来化简，因此得到：

$$
F^tF=\begin{array}{c}\\ 1\\ 2\\ 3\\ 4\end{array}\begin{array}{c}\begin{array}{cccc}1 & 2 & 3 & 4\end{array}\\ \left[\begin{array}{cccc}2 & 0 & 1 & 2\\ 0 & 2 & 2 & 1\\ 1 & 2 & 3 & 2\\ 2 & 1 & 2 & 3\end{array}\right]\end{array}
$$

考虑第一行，我们知道树支 2 是 T 的一个末梢树支。由于仅有两条路径包括了这条支路，因此不需要象12.4.3节设想的那样检验路径的一致性。划去第 2 列，并且由对应于 $p_1 \oplus p_2$ 的行来增广该矩阵，则得：

$$
F_1(\text{增广后的})=\begin{array}{c}\begin{array}{ccc}1 & 3 & 4\end{array}\\ \left[\begin{array}{ccc}1 & 1 & 1\\ 0 & 1 & 1\\ 0 & 1 & 0\\ 1 & 0 & 1\\ 0 & 0 & 1\end{array}\right]\end{array}
$$

划去对应于平凡路径的行之后，我们得到

$$
F'_1=\begin{array}{c}\begin{array}{ccc}1 & 3 & 4\end{array}\\ \left[\begin{array}{ccc}1 & 1 & 1\\ 0 & 1 & 1\\ 1 & 0 & 1\end{array}\right]\end{array}
$$

对于F'_1，W矩阵为

$$
\begin{array}{c}\\ 1\\ 3\\ 4\end{array}\begin{array}{c}\begin{array}{ccc}1 & 3 & 4\end{array}\\ \left[\begin{array}{ccc}2 & 1 & 2\\ 1 & 2 & 2\\ 2 & 2 & 3\end{array}\right]\end{array}
$$

检验第一行，我们得知，树支 3 是T_1的一个末梢树支。从F'_1中划去对应于 3 的列后所得到的实现F'_2的树 T_2 仅有两个支路，如图12.12(a)所示。对于这个树，我们加进树支 3，并使得{1,3,4}和{3,4}为路径，以及得到 T_1，如图12.12(b)所示。树支 2 现在再加到T_1，使得{2,3,4}和{2,3}为路径。这样，如图12.12(c)所示的树 T 就得到了。

作为第二个例子，考虑下面的矩阵

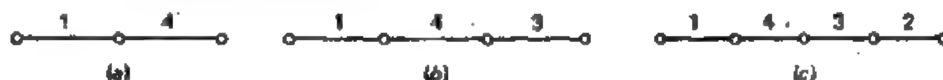


图 12.12 (a)树T'_2；(b)树T'_1；(c)树T。

$$F=\begin{array}{c}\begin{matrix}1 & 2 & 3 & 4\end{matrix}\\ \begin{bmatrix}1 & 0 & 1 & 0\\ 1 & 0 & 1 & 1\\ 1 & 1 & 1 & 1\\ 1 & 1 & 0 & 1\\ 1 & 1 & 0 & 0\end{bmatrix}\end{array}$$

求得相应的矩阵W为

$$W=\begin{array}{c|cccc|} & 1 & 2 & 3 & 4\\ \hline 1 & 5 & 3 & 3 & 3\\ 2 & 3 & 3 & 1 & 2\\ 3 & 3 & 1 & 3 & 2\\ 4 & 3 & 2 & 2 & 3\end{array}$$

对于行1， 集合$S=\{2,3,4\}$。因为这个集合S不可能再进一步化简了，所以我们从中划去任意一个元素，比如2。现在我们发现$w_{2,4}>w_{2,3}$，因此树支3是一个末梢树支，这样，$p_1=\{1\}$；$p_2=\{1,4\}$以及$p_3=\{1,2,4\}$，进而$p_1\oplus p_2\neq p_3$。我们现在从F中划去第3列，并且用$p_1\oplus p_2$、$p_1\oplus p_3$和$p_2\oplus p_3$行来增广它。在移去对应于平凡路径的行之后，我们得到的F'_1为

$$F'_1=\begin{array}{c}\begin{matrix}1 & 2 & 4\end{matrix}\\ \begin{bmatrix}1 & 0 & 1\\ 1 & 1 & 1\\ 1 & 1 & 0\\ 0 & 1 & 1\end{bmatrix}\end{array}$$

相应的W矩阵为

$$\begin{array}{c|ccc|} & 1 & 2 & 4\\ \hline 1 & 3 & 2 & 2\\ 2 & 2 & 3 & 2\\ 4 & 2 & 2 & 3\end{array}$$

相对于W的第一行的集合S是由2和4组成的。因为这个集合S不可能再进一步化简了，所以从中划去任意一个元素，比如2。因为在这一步中，S中仅存在一个元素，所以4是一个末梢树支。因此我们有$p_1=\{1\}$，$p_2=\{1,2\}$和$p_3=\{2\}$。由于p_1和p_2的环和等于p_3，所以矩阵F不可能实现。

12.5 （$n+1$）节点电阻n端口网络的实现——II

在12.3节中，我们提到有两种基本方法——一种由Cederbaum提出，另一种由Guillemin提出——实现一个已知实对称矩阵$Y=[y_{ij}]$作为一个（$n+1$）节点电阻n端口网络的短路电导矩阵问题，这个n端口网络不包含负电导。我们在12.3和12.4节已讨论过了Cederbaum方法中的两个主要步骤。

现在讨论 Rao 在文献[12.4]中给出的一个算法，它使用了 Guillemin 的方法。Guillemin方法象前面描述一样，由下面主要步骤构成：

1.求得实现矩阵Y的n端口网络N的端口图T。

2.根据Y和T，求得描述N的n端口网络的短路电导矩阵Y'，这里Y'有一个星-树端口图，其中星顶点对所有端口都是负的参考端点（见12.2.4节）。

3.如果矩阵Y'是超优势的，那么就能容易地获得所需要的网络（见12.2.2）；否则，Y就是不可能实现的。

12.5.1 末梢端口的识别

在Rao算法中，用来实现一个（$n+1$）节点电阻n端口网络的Y矩阵的头一个主要步骤，就是识别所需网络的端口图的末梢端口。我们容易地看到，应用12.4.2节所描述的矩阵$[y_{ij}]$算法，就能确定这样的末梢端口。为了完整起见，下面给出这个算法的全部步骤。

第一步 考虑Y的任意一行i。设m_i是这行中绝对值最小的元素，设S为列的集合，其中i行的元素为$\pm m_i$。如果S仅包含一个元素，那么这个元素就对应一个末梢端口。否则，设$S=\{s_1,s_2,\cdots,s_k\}$。

第二步 考虑Y的另一任意行j，这里$j\notin S$。如果对于S中的某些s_p和s_q，有$|y_{js_p}|>|y_{js_q}|$，那么就从S中删去s_p。并从j行中删去所有这样的s_p。

第三步 对其它某个行k（k不包含在化简了的集合S中），重复第二步。如果重复应用这个步骤而不能导致集合S仅包含一个元素，那么就从这个阶段的集合S中任意删去一个元素，并且再重复第二和第三步。最后S总能仅包含一个可识别为末梢端口的元素。

12.5.2 无零元素Y矩阵的实现

现在我们考虑不包含零元素的Y矩阵的实现。允许有零元素的一般情况，在[12.4]中已进行了论述。

实现全非零元素Y矩阵的网络N，是以存在一个生成N的每个端口对的连支为特征的。如果我们在T中指定一个从一个末梢顶点到另一个顶点的路径，作为最大路径，那么对于每个这样的最大路径，在N中都存在一个连支。进而，这个连支的电导等于最大路径的两个末梢端口间的传输电导值。下面，每个最大路径将用按顺序组成的端口序列识别和表示出来，序列的顺序下面用水平线表示。如果所有的或某些包含在路径中端口的确切顺序不知道，那么我们就用括号把相应的端口括起来。因此，例如$\underline{i、(j,\ k)、p、q、(r,s,t)}$是一条包含了$i$、$j$、$k$、$p$、$q$、$r$、$s$和$t$的最大路径，其中第二和第三位的$j$和$k$的确切位置以及在最后三位中的$r$、$s$和$t$的相对位置还有待确定。

设端口i用先前介绍的方法识别为一个末梢端口。我们的目的是根据T的末梢顶点确定最大路径集合。显然，这些有顺序的最大路径足以建立端口树的图。

设m_i是Y中第i行绝对值最小的元素，设传输电导y_{ij}（$j\neq i$，只是其中之一）等于$\pm m_i$。端口j可能是也可能不是末梢端口。当它不是末梢端口时，L集合（12.4节）的

端口（它不包含端口 i，并以 j 作为引导端口）形成了T的一个线性子树，并且在N中仅有一个生成 i 和L集合端口的连支。所以，不论 j 是不是末梢端口，在T中仅存在一条包含了 i 和 j 的最大路径。

设y_{ij}为正（负）；对于任意一个 k，如果y_{ik}和y_{jk}符号一致（不同），那么k就在包含 i 和 j 的最大路径中（定理12.2）。用这种方法，包含在最大路径中的所有端口都可被确定。设这个路径为 i（j，k，l，…），这里端口的顺序将在后面确定。很显然，在前面最大路径的表示中，括号里没有一个端口能是另外一个包括 i 的最大路径的末梢端口。为了帮助我们在下面紧接的步骤中不考虑这些端口，所以在Y中第 i 行中对应于列 j、k、l、…的各项用圆圈圈起来。我们现在选择另外一项 $y_{ip}(p\neq i)$，它在第 j 行中未圈起来的项中具有最小绝对值，并获得另外一条最大路径，即确定所有使 $y_{ip}y_{xi}y_{xp}$ 为正的x。设这个路径为i，（p，q，…）。我们现在把y_{ip}，y_{iq}，…这些元素圈起来。重复进行这个过程，直到Y的第 i 行中除了 y_{ii} 以外的所有元素都被圈起来为止。我们注意到，每个端口 j、p、…仅在一个最大路径中引入，而其它的端口可在多于一个最大路径中引入。

如果我们把每个这样的最大路径的端口按它们出现的顺序排列，那么构造树所需数据就是完整的。为此，我们将每个最大路径中的端口，按它们对端口 i 的传输电导值的递减顺序排列。当不止一个端口与端口 i 的传输电导有相同值时，我们用括号把这些端口括起来。因此，当第一步完成后，我们就有了局部排序序列的集合：

$$\underline{i,\ m,\ (h,\ k,\ n,\ o),\ s,\ (t,\ u,\ v),\ (w,\ x,\ j).}$$

现在必须把每一组括号里的端口排序。为此，我们考虑相应于端口m的行，这个端口刚好位于包含h、k、n和o括号的前面，并且把括号内的这些端口，按照它们相对于端口m的传输电导值的递减顺序排列。如果某些传输电导再次出现相等的值，则仍然用括号把这些端口括起来并重复上面的过程。端口排序也可由考虑紧跟在括号后的端口 s 得出，这时端口顺序按照它们对端口 s 传输电导值的递增顺序排列。用这种方法，就能确定每一个最大路径中的端口顺序。

然而，有时也可能遇到这种情况，即括号里的所有端口对外面的任一端口的传输电导都具有相同值。例如，对于序列…、$\underline{s、(p,\ q,\ r)、t\cdots}$，我们可能有

$$|y_{sp}| = |y_{sq}| = |y_{sr}| \text{ 和 } |y_{pt}| = |y_{qt}| = |y_{rt}|$$

在这种情况下，我们抽出相应于括号中这些端口的Y的方子矩阵，并且分别确定它的线性-树端口网络。设q、r、p为这样确定的顺序，那么$\underline{\cdots,\ s,\ q,\ r,\ p,\ t,\cdots}$或者$\underline{\cdots,\ s,\ p,\ r,\ q,\ t,\ \cdots}$都可作为排序序列，并且两个相应的图是2-同构的。

在一个最大路径中的路径排序完毕之后，可实现矩阵的其他最大路径中的共有支路排序就和已有的排序相同。在一个新的路径中排序支路时，也可以借助于这个事实。

下面我们通过实现10阶电导矩阵Y的端口-树图来说明上述过程：

$$
Y=\begin{array}{c}
 \\ 1\\2\\3\\4\\5\\6\\7\\8\\9\\10
\end{array}
\begin{array}{cccccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10
\end{array}
$$

$$
Y=\left[\begin{array}{rrrrrrrrrr}
67 & 20 & 10 & 15 & 45 & 17 & 23 & 33 & 15 & 26\\
20 & 30 & 3 & -6 & 23 & 5 & 12 & 8 & -3 & 29\\
10 & 3 & 30 & 2 & 5 & -4 & -1 & 11 & 5 & 3\\
15 & -6 & 2 & 45 & 30 & 9 & 4 & 11 & -15 & -12\\
45 & 23 & 5 & 30 & 79 & 14 & 16 & 25 & -18 & 29\\
17 & 5 & -4 & 9 & 14 & 30 & -7 & 24 & 3 & 5\\
23 & 12 & -1 & 4 & 16 & -7 & 49 & -8 & 7 & 12\\
33 & 8 & 11 & 11 & 25 & 24 & -8 & 42 & 8 & 14\\
15 & -3 & 5 & -15 & -18 & 3 & 7 & 8 & 36 & -3\\
26 & 29 & 3 & -12 & 29 & 5 & 12 & 14 & -3 & 41
\end{array}\right]
$$

（行号与列号均为 1, 2, …, 10。）

1.考虑第一行，我们确定端口 3 为末梢端口。

2.在第 3 行中，最小值元素在第 7 列位置上，且 y_{37}是负的。扫视一下第 3 、 7 列，我们注意到，对于 $k=3$、7 和 8，y_{k3}和y_{k7}具有相反的符号。因此，3，(7，8)是最大路径。在第 3 行中，我们圈住第 7 列和第 8 列的元素，则变为

$$[10 \quad 3 \quad 30 \quad 2 \quad 5 \quad -4 \quad -1 \quad ⑪ \quad 5 \quad 3]$$

3.在第 3 行没有圈住的元素中，y_{34}具有最小值，并且为正。扫视一下第 3 、4 列，我们注意到，对于k=1、3、4、5和8，y_{k3}和y_{k4}具有相同的符号，于是 3，(1，4，5，8)是另一个最大路径。现在我们已圈住了y_{31}、y_{34}和 y_{35}项。类似地，我们还可得到其它的最大路径 3，(2，1，5，8，10)；3，6和3，(9，1，8)。

4.考虑路径 3，(2，1，5，8，10)。将括号中的端口按其对端口 3 的传输电导值的递减顺序排列，则得

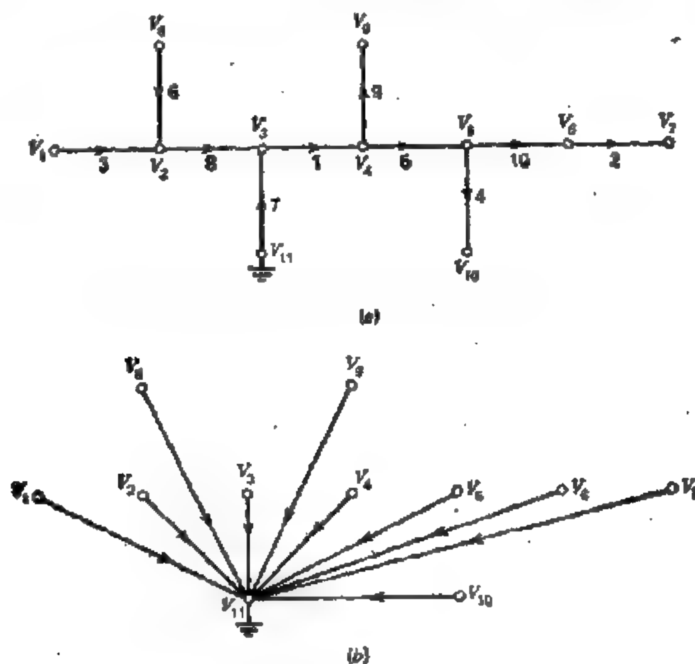


图 12.13 (a)树T；(b)树*。

$$\underline{3,\ 8,\ 1,\ 5,(2,\ 10)}$$

现在考虑 Y 的第5行，我们发现 $|y_{5,10}|>|y_{5,2}|$。因此，$\underline{3,\ 8,\ 1,\ 5,\ 10,\ 2}$ 是一个有序的最大路径。

5.因为其它路径和这个路径有公共支路，所以它们很容易来排列。因此，其它有序路径是 $\underline{3,\ 8,\ 7}$；$\underline{3,\ 8,\ 1,\ 5,\ 4}$ 和 $\underline{3,\ 8,\ 1,\ 9}$。

6.现在端口树 T 可以很容易地确定了，如图12.13(a)所示。

7.任意一个端口的方向（比如端口1）都能任意确定。其余端口的方向要用定理12.1来确定。最后所得到的端口方向，如图12.13（a）所示。

8.设与 T 有同样顶点集合的星树 T^* 的选择，如图12.13（b）所示。

关联端口电压矢量 V_p 和 V_p^* 的矩阵 k 给出如下：（注意 $V_p=kV_p^*$）

$$K=\begin{bmatrix} 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

对应于 T^* 的短路电导矩阵 Y^* 给出为

$Y^*=K^tYK$

$$=\begin{bmatrix} 30 & -15 & 0 & 0 & 0 & 0 & -3 & -4 & -5 & -2 \\ -15 & 24 & -1 & 0 & 0 & -6 & 0 & -2 & 0 & 0 \\ 0 & -1 & 30 & -7 & -4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -7 & 26 & -16 & 0 & 0 & 0 & -3 & 0 \\ 0 & 0 & -4 & -16 & 23 & 0 & 0 & 0 & 0 & -3 \\ 0 & -6 & 0 & 0 & 0 & 13 & -1 & 0 & 0 & -6 \\ -3 & 0 & 0 & 0 & 0 & -1 & 30 & -5 & -3 & -6 \\ -4 & -2 & 0 & 0 & 0 & 0 & -5 & 30 & -3 & -9 \\ -5 & 0 & 0 & -3 & 0 & 0 & -3 & -3 & 36 & -15 \\ -2 & 0 & 0 & 0 & -3 & -6 & -6 & -9 & -15 & 45 \end{bmatrix}$$

矩阵 Y^* 是超优势的。实现 Y 的10端口网络的电导，可由 Y^* 计算出来，并由下面的关系给出：

$$g_{ij}=-y_{ij},\qquad 1\leqslant i,\ j\leqslant 10$$

$$g_{i,11}=y_{ii}-\sum_{j\neq i}^{10}|y_{ij}|,\qquad 1\leqslant i\leqslant 10$$

12.6 进一步阅读

研究n端口电阻网络的领域主要涉及到如下几个题目：

1. $n+1$个节点的n端口网络的Y矩阵，即秩为n的n端口网络。

2. 多于$n+1$个节点的n端口网络的Y矩阵。

3. 零度为n的n端口网络的Z矩阵。

为了实现割集和回路矩阵，在12.3节给出了Cedesbaum分解算法以及在12.4节中给出了Rao算法，两种算法结合起来就完全解决了构造具有给定Y矩阵的$n+1$个节点n端口网络的问题。为解决这个问题，Rao文献在[12.4]还给出了另一个替代算法。这个算法，正如我们在12.5节中提到的，是基于Guillemin方法的，它包括了根据已知Y矩阵构造出端口图。关于实现无非零元素Y矩阵的算法，参见文献[12.5]。早期关于$n+1$个节点电阻n端口网络的论文，包括在文献[12.6]和[12.7]中。

Guillemin文献[12.8]给出了实际具有大于$n+1$个节点的n端口网络Y矩阵问题的一般研究方法。使用这个方法，在试图发现这种网络的Y矩阵的性质方面作了许多尝试，但是，除了在$(n+2)$个节点网络的特殊情况以外，没有得到很多有意义的结果。请参看文献[12.9]、[12.10]以及[12.11]。

如同我们在12.3节看到的那样，Cederbaum算法可以用来分解零度为n的n端口网络的Z矩阵，这与12.4节中的算法合起来就完全解决了实现零度为n的n端口网络的开路电阻矩阵的问题。解决这个问题的其它方法，在文献[12.12]和[12.13]中都有所讨论。也请参见文献[12.14]和[12.15]，这里讨论了电阻网络和通讯网之间的关系。

Cerderbaum在文献[12.16]引入了改进的割集和回路矩阵的概念。关于这类矩阵在研究n端口网络问题中的应用，请参见文献[12.17]和[12.18]。

很多关于n端口电阻网络的其他有趣问题已在文献中有所讨论。例如，参见[12.19]，[12.20]以及[12.21]。

关于实现割集和回路矩阵的问题，请参见文献[12.22]，[12.23][12.24]和[12.25]。

文献[12.26]讨论了电阻n端口网络领域中的几个结果，并且还讨论了实现回路矩阵在开关网络综合中的应用。由电阻n端口网络领域建立起来的概念，能够用来帮助实现RLC网络的状态矩阵，请参见文献[12.27]。

12.7 习　　题

12.1 考虑n端口网络N，假定不存在仅由端口组成的回路。如果I_p是端口电流向量，I是与N中非端口元素所对应的电流向量，证明：存在元素值为1、或-1、或0的矩阵M，使得$I_p=MI$。

提示：利用定理10.31（见文献[12.28]）。

12.2 用割集代替回路，V_p代替I_p，V代替I，重作习题12.1。其中V_p和V分别是与N的端口元素和非端口元素所对应的向量。

12.3 用Cederbaum方法找出具有如下短路电导矩阵的电阻6端口网络：

$$\begin{bmatrix} 3 & 2 & 0 & 1 & -1 & 2 \\ 2 & 5 & -2 & 0 & 2 & 2 \\ 0 & -2 & 9 & 0 & -3 & 0 \\ 1 & 0 & 0 & 9 & -1 & -1 \\ -1 & 2 & -3 & -1 & 8 & 0 \\ 2 & 2 & 0 & -1 & 0 & 8 \end{bmatrix}$$

12.4 实现下面的树-路矩阵向量所表示的图：

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

12.5 设$Z[z_{ij}]$是零度为n的n端口网络N的开路电阻矩阵。证明：如果N的端口构成了增广网络$\hat{N}$的一个生成星树，那么Z满足：对于每三个下标i、j、k，或者是所有三个元素z_{ij}、z_{jk}、z_{ki}的数值相等，或者是两个数值较小的相等（参见文献[12.14]）。

12.6 令与n端口网络N相关的矩阵Q_1、Q_2、Y_{12}、Y_{22}等，其定义如12.1节中所述。矩阵$Q_1-Y_{12}Y_{22}^{-1}Q_2$叫作N的修改割集矩阵。证明：

(a) $Y=Q_mY_bQ_m^t$，这里Q_m是N的修改割集矩阵。

(b) 矩阵Q_m是n端口网络N的修改割集矩阵，当且仅当$Q_mY_bQ_2^t=0$。

12.7 令N是两个n端口网络N_1和N_2的并联组合，N_1和N_2的短路电导矩阵分别为Y_1和Y_2，证明：N的短路电导矩阵等于Y_1+Y_2，当且仅当N_1和N_2有相同的修改割集矩阵（参见文献[12.18]）。

12.8 运用习题12.6和12.7的结果，构造出一个4端口网络，它的短路电导矩阵等于图12.14所示的4端口网络的短路电导矩阵，图中所有电导以西门子为单位。

图 12.14

12.9 令与n端口网络N相关的矩阵B_1、B_2、Z_{11}、Z_{12}等，其定义如12.1节中所述。那么矩阵（$B_1-Z_{12}Z_{22}^{-1}B_2$）叫作N的修改回路矩阵。证明：

（a）$Z=B_mZ_eB_1^t$，这里B_m是N的修改回路矩阵。

（b）矩阵B_m是n端口网络N的修改回路矩阵，当且仅当$B_mZ_eB_2^t=0$。

12.10 考虑两个具有相同的端口图、边图和方向的n端口网络N_1和N_2。令Z_1和Z_2是N_1和N_2的开路阻抗矩阵。现在来构造第三个具有与N_1和N_2相同的边图、端口图和方向、但每边的阻抗是N_1和N_2对应边阻抗之和的n端口网络N_3。网络N_3称作N_1和N_2的伪-串联组合（网络N_1和N_2以及它们的伪-串联组合N_3，如图12.15所示）。证明：N_3的开路阻抗矩阵等于Z_1+Z_2，当且仅当N_1和N_2都有相同的修改回路矩阵（参见文献[12.29]）。

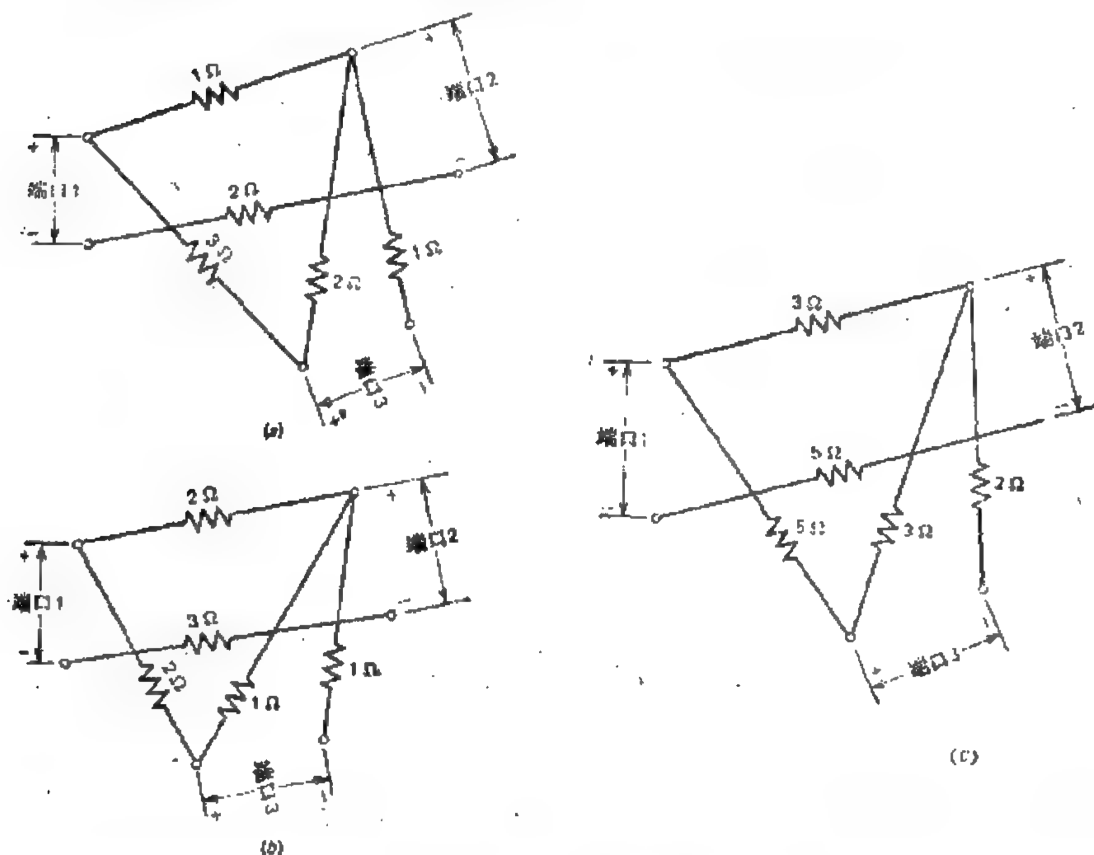


图 12.15 (a)N_1；(b)N_2；(c)N_1和N_2的伪-串联组合。

12.11 运用习题12.9（b）的结果，构造出一个4端口网络，它的修改回路矩阵与图12.14所示网络的相同。

12.12 给定一个n端口网络N，运用习题12.9和12.10的结果，试构造一个新的4端口网络，它的开路阻抗矩阵与图12.14所示网络的相等。讨论所碰到的各种问题。

12.8 参考文献

12.1 I Cederbaum, "Applications of Matrix Algebra to Network Theory," *IRE Trans Circuit Theory* (special supplement), Vol. CT-6, 127—137(1959)

12.2 L Weinberg, *Network Analysis and Synthesis*, McGraw-Hill, New York, 1962.

12.3 E. A Guillemin, "On the Analysis and Synthesis of Single-Element Kind Ne-

tworks," *IRE Trans. Circuit Theory,* Vol CT-7, 303—312(1960).

12.4 V. V. Bapeswara Rao, "The Tree-Path Matrix of a Network and Its Applications," *Ph. D. Thesis,* Dept. of Electrical Engineering, Indian Institute of Technology, Madras, India, 1970.

12.5 F. T. Boesch and D. C. Youla, "Synthesis of $(n+1)$-Node Resistor n-Ports," *IEEE Trans. Circuit Theory,* Vol. CT-12, 515—520(1965).

12.6 G. Biorci and P. P. Civalleri, "On the Synthesis of Resistive N-Port Networks," *IRE Trans. Circuit Theory,* Vol CT-8, 22—28(1961).

12.7 C. C. Halkias, I. Cederbaum, and W. H. Kim, "Synthesis of Resistive n-Ports with$(n+1)$ Nodes," *IRE Trans. Circuit Theory,* Vol. CT-9, 69—73(1962).

12.8 E.A. Guillemin,"On the Realization of an n-th Order G-Matrix," *IRE Trans. Circuit Theory,* Vol. CT-8, 318—323(1961).

12.9 K.R. Swaminathan and I. T. Frisch,"Necessary Conditions for the Realizability of n-Port Resistive Networks with more than$(n+1)$ Nodes," *IEEE Trans. Circuit Theory,* Vol. CT-12, 520—527(1965).

12.10 G. Biorci and P.P. Civalleri, "Analysis of Resistive N-Port Networks Based on$(n+2)$ Nodes," in *Aspects of Network and System Theory* (Eds. R.E.Kalman and N. DeClaris), Holt, Rinehart, Winston, New York, 1971.

12.11 P. Subbarami Reddy and K. Thulasiraman,"Synthesis of$(n+2)$-Node Resistive n-Port Networks," *IEEE Trans. Circuit Theory,* Vol. CT-19, 20—25(1972).

12.12 D.P. Brown,"N-Port Synthesis of N-order Positive Entry Resistance Matrices," *J. Franklin Inst.,* Vol. 284, 26—38(1967).

12.13 C. Eswaran and V. G. K. Murti, "Realization of Positive Entry Resistance Matrices," *AEU,* Vol. 29, 212—216(1975).

12.14 C.Eswaran and V. G. K.Murti,"On a Relationship between Terminal Capacity and Impedance Matrices," *IEEE Trans. Circuits and Sys.,* Vol. CAS-21, 732—734(1974).

12.15 I. T. Frisch and W. H. Kim, "n-Port Resistive Networks and Communication Nets," *IRE Trans. Circuit Theory,* Vol. CT-8, 493—496(1961).

12.16 I. Cederbaum, "On Equivalence of Resistive N-Port Networks," *IEEE Trans. Circuit Theory,* Vol. CT-12, 338—344(1965).

12.17 P. Subbarami Reddy, V.G.K. Murti, and K. Thulasiraman, "Realization of Modified Cutset Matrix and Applications," *IEEE Trans. Circuit Theory,* Vol. CT-17, 475—486(1970).

12.18 A. Lempel and I. Cederbaum, "Parallel Interconnection of n-port Networks," *IEEE Trans. Circuit Theory,* Vol. CT-14, 274—279(1967).

12.19 A. Lempel and I. Cederbaum, "Terminal Configurations of n-Port Networks," *IEEE Trans. Circuit Theory,* Vol. CT-15, 51—53(1968).

12.20 S. Prigozy and L.Weinberg,"Realization of Fourth-Order Singular and Quasi-Singular Resistance and Conductance Matrices," *IEEE Trans. Circuits and Sys.,* Vol. CAS-23, 245—253(1976).

12.21 M. C. Govindaralulu Naidu, P. Subbarami Reddy, and K. Thulasiraman, "$(n+2)$-Node Resistive n-Port Realizability of Y-Matrices," *IEEE Trans. Circuits and Sys.*, Vol. CAS-23, 254—261(1976).

12.22 W.T. Tutte, "An Algorithm for Determining whether a Given Binary Matroid is Graphic," *Proc. Am. Math. Soc.*, Vol. 11, 905—917(1960).

12.23 W. T. Tutte, "From Matrices to Graphs," *Can. J. Math.*, Vol. 56,108—127(1964).

12.24 W. Mayeda, "Necessary and Sufficient Conditions for the Realizability of Cut-set and Circuit Matrices," *IRE Trans. Circuit Theory*, Vol. CT-7, 79—81(1960).

12.25 W. Mayeda, "A Proof of Tutte's Realizability Condition," *IEEE Trans. Circuit Theory*, Vol. CT-17, 506—511(1970).

12.26 W.H. Kim and R.T. Chien, *Topological Analysis and Synthesis of Communication Networks*, Columbia Univ. Press, New York, 1962.

12.27 M.N.S. Swamy and K. Thulasiraman, "Realization of the A-Matrix of RLC Networks," *IEEE Trans. Circuit Theory*, Vol. CT-19, 515—518(1972).

12.28 L.O. Chua and D.N. Green, "Graph-Theoretic Properties of Dynamic Nonlinear Networks," *IEEE Trans. Circuits and Sys.*, Vol. CAS-23, 292—302(1976).

12.29 K. Thulasiraman and V. G. K. Murti, "Pseudo-Series Combination of n-Port Networks," *Proc. IEEE*, Vol. 56, 1143—1144(1968).

第十三章　网络函数和网络灵敏度

本章，我们首先根据与一给定网络的某些子图有关的导纳来推导网络函数的公式，这些公式（或称拓扑公式）是1847年首先由基尔霍夫以电阻形式给出的，而后又于1892年由麦克斯韦以导纳形式给出的。本章的推导是根据不定导纳矩阵，即将网络函数作为不定导纳矩阵相应余子式之比来求得拓扑公式。因而本章的许多结论很容易从第六章的结论导出。

在本章后半部分，我们将介绍一个计算网络函数灵敏度的方法，此一种方法基于伴随网络的概念和特勒根定理（定理11.3）。

13.1　无互感RLC网络的拓扑公式

本节，我们先从节点导纳矩阵入手，推导出无互感RLC网络的拓扑公式。

首先考虑一个单口无互感RLC网络N。设N具有$n+1$个节点，记为0、1、2、…、n，并设节点1和0分别为端口的正、负参考端子（图13.1）。假定N中全部初始条件（电容电压和电感电流）均为零。进而，电压和电流变量都是复频变量的拉普拉斯变换。现在，让我们在端口上跨接一个值为I_1的电流源来激励此网络。如果V_1、V_2…V_n表示节点1、2、…n相对于节点0的电压，Y表示网络N以0为参考点的节点导纳矩阵，那么N的节点方程为

$$YV=I \tag{13.1}$$

这里，

$$V=\begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_n \end{bmatrix}, \quad I=\begin{bmatrix} I_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

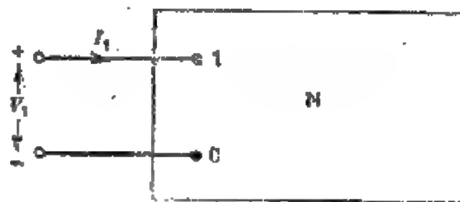


图 13.1　单口网络

从式（13.1）解出V_1则得

$$V_1=\frac{\Delta_{11}}{\Delta}I_1$$

这里，

$$\Delta=\det Y,\ \Delta_{11}=Y\text{的（1，1）余子式}$$

因此，N的策动点阻抗z为

$$z=\frac{V_1}{I_1}=\frac{\Delta_{11}}{\Delta} \tag{13.2}$$

而且N的策动点导纳y为

$$y=\frac{1}{z_1}=\frac{\Delta}{\Delta_{11}} \tag{13.3}$$

为了导出z、y的拓扑公式，我们必须用与N的某些子图有关的适当量来描述Δ_{11}和

Δ，这在下述方法中很容易做到。

首先，我们定义N的一个子图的导纳积为与该子图各边相应的导纳之积．如果一个子图没有边，那么它的导纳积定义为１．N的一个子图的阻抗积也由类似的方法来定义。

运用这些定义，令

W = N的全部生成树的导纳积之和，

$W_{i,j}$ = N的全部生成2-树的导纳积之和，

$C(W)$ = N的全部生成余树的阻抗积之和，

$C(W_{i,j})$ = N的全部生成2-树$T_{i,j}$的补的阻抗积之和　　(13.4)

如果z_1，z_2，$\cdots z_m$是N中各边的阻抗，那么很显然

$$C(W)=W\prod_{i=1}^{m}z_i \tag{13.5}$$

$$C(W_{i,j})=W_{i,j}\prod_{i=1}^{m}z_i \tag{13.6}$$

现在讨论N的节点导纳矩阵。如果将N视为一个以导纳来表示相应边权的赋权图，那么我们就会发现（习题6.16）

$$\Delta=\det Y=W \tag{13.7}$$

$$\Delta_{i,j}=W_{ij,0} \tag{13.8}$$

将式（13.5）到（13.8）代入式（13.2）和（13.3），则得如下定理。

定理13.1　设y、z分别表示一个单口无互感RLC网络的策动点导纳和策动点阻抗．如果1和0是该网络的口端子，那么

1.　$$z=\frac{W_{1,0}}{W}=\frac{C(W_{1,0})}{C(W)}$$

2.　$$y=\frac{W}{W_{1,0}}=\frac{C(W)}{C(W_{1,0})}$$　□

下面我们推导双口无互感RLC网络N（图13.2)的开路阻抗和短路导纳函数的拓扑公式．再次假定N中全部初始条件均为零。若以值为I_1、I_2的电流源来激励N的双口网络，那么N的节点方程可写为

$$YV=I$$

这里

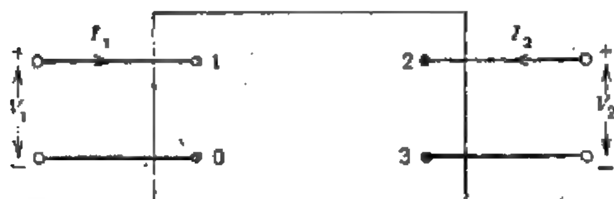


图 13.2　双口网络

$$I=\begin{bmatrix} I_1 \\ I_2 \\ -I_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

解出节点电压V_1、V_2、V_3，则有

$$V_1=\frac{1}{\Delta}(\Delta_{11}I_1+\Delta_{21}I_2-\Delta_{31}I_2)$$

$$V_2=\frac{1}{\Delta}(\Delta_{12}I_1+\Delta_{22}I_2-\Delta_{32}I_2)$$

$$V_3=\frac{1}{\Delta}(\Delta_{13}I_1+\Delta_{23}I_2-\Delta_{33}I_2)$$

从上述关系可得

$$\begin{bmatrix} V_1 \\ V_2-V_3 \end{bmatrix}=\frac{1}{\Delta}\begin{bmatrix} \Delta_{11} & \Delta_{21}-\Delta_{31} \\ \Delta_{12}-\Delta_{13} & \Delta_{22}+\Delta_{33}-\Delta_{32}-\Delta_{23} \end{bmatrix}\begin{bmatrix} I_1 \\ I_2 \end{bmatrix} \tag{13.9}$$

注意：因为Y是对称的，所以有

$$\Delta_{ij}=\Delta_{ji}$$

由于V_1是端口 1 的电压，V_2-V_3是端口 2 的电压，所以式（13.9）中的系数矩阵就等于N的开路阻抗矩阵Z_{oc}。即

$$Z_{oc}=\frac{1}{\Delta}\begin{bmatrix} \Delta_{11} & \Delta_{12}-\Delta_{13} \\ \Delta_{12}-\Delta_{13} & \Delta_{22}+\Delta_{33}-2\Delta_{23} \end{bmatrix} \tag{13.10}$$

为了用N中恰当的导纳积的形式来描述Z_{oc}的元素，先观察得到

$$\Delta_{11}=W_{1,0} \tag{13.11}$$

再由式（13.8）得

$$\Delta_{12}-\Delta_{13}=W_{12,0}-W_{13,0} \tag{13.12}$$

因为每棵生成2-树$T_{12,0}$或者是$T_{12,30}$，或者是$T_{123,0}$，则有

$$W_{12,0}=W_{12,30}+W_{123,0} \tag{13.13}$$

类似地，

$$W_{13,0}=W_{13,20}+W_{123,0} \tag{13.14}$$

把式（13.13）和（13.14）代入式（13.12），则得

$$\Delta_{12}-\Delta_{13}=W_{12,30}-W_{13,20} \tag{13.15}$$

同样道理，

$$\begin{aligned}\Delta_{22}+\Delta_{33}-2\Delta_{23}&=W_{2,0}+W_{3,0}-2W_{23,0}\\&=W_{23,0}+W_{2,30}+W_{23,0}+W_{3,20}-2W_{23,0}\\&=W_{2,30}+W_{3,20}\\&=W_{2,3}\end{aligned} \tag{13.16}$$

将式（13.15）和（13.16）代入式（13.10），则得如下定理。

定理13.2 设N是一个双口无互感RLC网络，它的各口正，负端子如图13.2所示，那么N的开路阻抗矩阵Z_{oc}为

$$Z_{oc}=\frac{1}{W}\begin{bmatrix} W_{1,0} & W_{12,30}-W_{13,20} \\ W_{12,30}-W_{13,20} & W_{2,3} \end{bmatrix}$$ □

由式（13.5）、（13.6）可清楚地看到，Z_{oc}的元素也能够以适当的生成余树和生成余2-树的阻抗积来表示。

短路导纳矩阵Y_{sc}可由Z_{oc}之逆求得，即

$$Y_{sc}=\frac{1}{\Delta_{11,22}+\Delta_{11,33}-2\Delta_{11,23}}\begin{bmatrix}\Delta_{22}+\Delta_{33}-2\Delta_{23} & \Delta_{13}-\Delta_{12}\\ \Delta_{13}-\Delta_{12} & \Delta_{11}\end{bmatrix} \tag{13.17}$$

这里应用了恒等式

$$\Delta_{ii,jk}=\frac{1}{\Delta}(\Delta_{ii}\cdot\Delta_{jk}-\Delta_{ji}\cdot\Delta_{ik}) \tag{13.18}$$

注意，$\Delta_{ii,jk}$是Y中对应（i，i）和（j，k）元素的二阶余子式，并表示为

$$\Delta_{ii,jk}=\mathrm{sgn}(i-j)\mathrm{sgn}(i-k)(-1)^{j+k}\det(Y_{ii,jk}) \tag{13.19}$$

这里：

1.$\mathrm{sgn}(r-s)=\begin{cases}1, & r-s>0\\ -1, & r-s<0\end{cases}$

2.$Y_{ii,jk}$是从Y中删去第i、j行和第i、k列而得到的矩阵。实际上，$\Delta_{ii,jk}$是Y_{ii}对应于Y中(j，k)元素的余子式，而且式（13.18）可称为雅可比等式[13.1]。

现在我们用适当的导纳积形式来表示$\Delta_{ii,jk}$。

考虑把N的节点i与0短路后所得到的网络N'。那么以0为参考点的N'的节点导纳矩阵Y'就等于Y_{ii}。如果用Δ'_{jk}表示Y'对应Y中（j，k）元素的余子式，那么如前所述

$$\Delta'_{jk}=\Delta_{ii,jk}$$

如果用$T'_{jk,0}$表示N'的一个生成2-树，其中节点j和k在一片上而节点0在另一片上，并且$W'_{jk,0}$表示N'中全部生成2-树$T'_{jk,0}$的导纳积之和，那么根据式（13.8），则得

$$\Delta'_{jk}=W'_{jk,0}$$

但是因为在N'中，节点0和节点i是同一点，所以N'的一个生成2-树$T'_{jk,0}$同时又是N中一个形如$T_{i,jk,0}$的生成3-树，即

$$\Delta_{ii,jk}=\Delta'_{jk}=W'_{jk,0}=W_{i,jk,0} \tag{13.20}$$

这里，$W_{i,jk,0}$是形如$T_{i,jk,0}$的全部生成3-树的导纳积之和。

接下来，考虑式（13.17）中的$\Delta_{11,22}+\Delta_{11,33}-2\Delta_{11,23}$项。由式（13.20）可见

$$\Delta_{11,22}+\Delta_{11,33}-2\Delta_{11,23}=W_{1,2,0}+W_{1,3,0}-2W_{1,23,0}$$

因为

$$W_{1,2,0}=W_{13,2,0}+W_{1,23,0}+W_{1,2,30}$$

和

$$W_{1,3,0}=W_{12,3,0}+W_{1,23,0}+W_{1,3,20}$$

所以得

$$\Delta_{11,22}+\Delta_{11,33}-2\Delta_{11,23}=W_{13,2,0}+W_{1,2,30}+W_{12,3,0}+W_{1,3,20} \tag{13.21}$$

根据式（13.8）、（13.15）、（13.16）和（13.21），就可得到Y_{sc}中全部元素的拓扑公式。

定理13.3 设N是一个双口无互感RLC网络，其各端口的正、负参考端子如图13.2所示，那么N的短路导纳矩阵Y_{sc}为

$$Y_{sc}=\frac{1}{W_{13,2,0}+W_{1,2,30}+W_{12,3,0}+W_{1,3,20}}\times$$

$$\times\begin{bmatrix} W_{2,3} & W_{13,20}-W_{12,30} \\ W_{13,20}-W_{12,30} & W_{1,0} \end{bmatrix}$$ □

现在我们来说明一个双口RLC网络的开路阻抗和短路导纳矩阵的拓扑计算方法。下面的例子选自文献[13.2]。

参见图13.3所示网络。该网络元件用符号a、b、c、d、e来表示，元件数值已在图中标出。

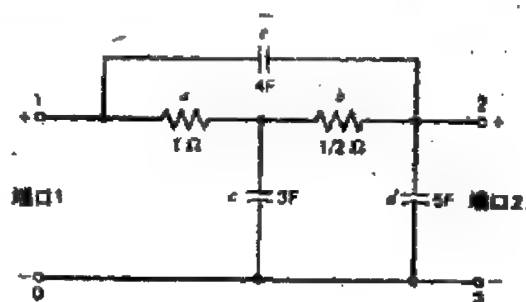


图 13.3

注意：3和0是同一顶点。因此，N不含有形如$T_{13,20}$的生成2-树。同理，N也不含有形如$T_{13,2,0}$、$T_{12,3,0}$、$T_{1,3,20}$的生成3-树。

用来求解Z_{oc}和Y_{sc}的生成树、2-树、3-树如下：

生成树：

(a, c, e)，(c, b, e)，(b, d, e)，(a, b, c)，

(a, b, d)，(a, c, d)，(a, d, e)和(c, d, e)

$T_{1,0}$：

(b, c)，(b, d)，(c, d)，(a, d)，

(a, b)，(b, e)，(c, e)和(a, e)

$T_{2,3}$：

(a, b)，(a, e)，(a, c)，(c, e)和(b, e)

$T_{12,30}$：

(a, b)，(a, e)，(b, e)和(c, e)

$T_{1,2,30}$：

(a)，(b)和(c)

由上可得W、$W_{1,0}$、$W_{2,3}$、$W_{12,30}$和$W_{1,2,30}$如下：

$$W=60s^3+111s^2+16s$$
$$W_{1,0}=27s^2+33s+2$$
$$W_{2,3}=12s^2+15s+2$$
$$W_{12,30}=12s^2+12s+2$$
$$W_{1,2,30}=3s+3$$

运用这些式子以及定理13.2和定理13.3，即可求出Z_{oc}和Y_{sc}

$$Z_{oc}=\frac{1}{60s^3+111s^2+16s}\times\begin{bmatrix} 27s^2+33s+2 & 12s^2+12s+2 \\ 12s^2+12s+2 & 12s^2+15s+2 \end{bmatrix}$$

$$Y_{sc}=\frac{1}{3s+3}\begin{bmatrix} 12s^2+15s+2 & -(12s^2+12s+2) \\ -(12s^2+12s+2) & 27s^2+33s+2 \end{bmatrix}$$

13.2 一般线性网络的拓扑公式

本节将讨论一般集中参数线性时不变网络的拓扑公式。我们将采用对应某给定网络的有向图中的有向树和有向2-树的导纳积来表达这些公式。推导这些公式的起点是如下定义的不定导纳矩阵。

考虑一个具有 n 个端子 1、2、…n 的网络N。令V_1、V_2、…V_n表示这些端子相对于网络外部任一端点 0 的电压，I_1，I_2…I_n表示网络与外部接通后、流入这些端子的电流（图13.4）。

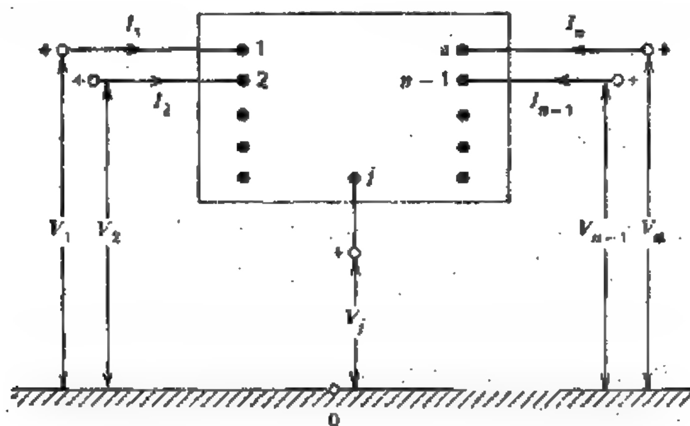


图 13.4 n 端子网络

表示N的端电压和电流关系的矩阵$\hat{Y}$，称为N的不定导纳矩阵。因此，

$$\hat{Y}V=I$$

即

$$\begin{pmatrix} \hat{y}_{11} & \hat{y}_{12} & \cdots & \hat{y}_{1n} \\ \hat{y}_{21} & \hat{y}_{22} & \cdots & \hat{y}_{2n} \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ \hat{y}_{n1} & \hat{y}_{n2} & \cdots & \hat{y}_{nn} \end{pmatrix} \begin{pmatrix} V_1 \\ V_2 \\ \cdot \\ \cdot \\ \cdot \\ V_n \end{pmatrix} = \begin{pmatrix} I_1 \\ I_2 \\ \cdot \\ \cdot \\ \cdot \\ I_n \end{pmatrix} \tag{13.22}$$

如果我们把网络N视为一个 n 口网络，其 i 端子与外部端子 0 形成第 i 口，那么就会发现，不定导纳矩阵$\hat{Y}$是这样一个 n 口网络的短路导纳矩阵。从而

$$\hat{y}_{jk}=\left.\frac{I_j}{V_k}\right|_{\text{全部}V_i=0,\,i\neq k}$$

换言之，如果我们把一个单位电压源跨接在端子k和外部端子 0 之间，并把其它端子与外部端子短接，那么流入端子j的电流I_j值就等于$\hat{y}_{jk}$。我们能够用这一方法来计算$\hat{Y}$的全部元素。例如，图13.5所示网络的不定导纳矩阵可证明等于

$$
\begin{bmatrix} 1+y & -y & -1 & 0 \\ -y & 2+y & -1 & -1 \\ -1 & -1-g_m & 2+g_p+g_m & -g_p \\ 0 & -1+g_m & -g_p-g_m & 1+g_p \end{bmatrix} \quad (13.23)
$$

该例选自文献[13.3]。

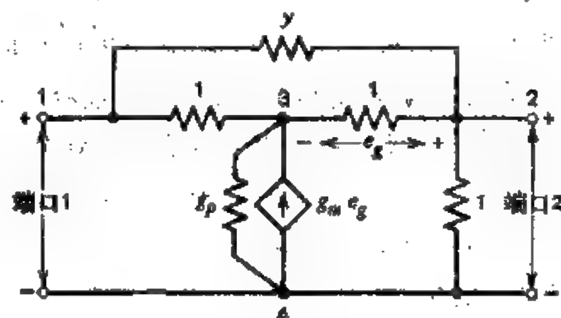


图 13.5

以下将证实不定导纳矩阵的两个有用性质。

根据基尔霍夫电流定律，

$$I_1+I_2+\cdots I_n=0$$

因此

$$(\hat{y}_{11}+\hat{y}_{21}+\cdots+\hat{y}_{n1})V_1+(\hat{y}_{12}+\hat{y}_{22}+\cdots+\hat{y}_{n2})V_2$$
$$+\cdots+(\hat{y}_{1n}+\hat{y}_{2n}+\cdots+\hat{y}_{nn})V_n=0$$

注意：电压V_1、V_2、$\cdots V_n$都是独立的。假定对所有$i\neq k$，$V_i=0$。那么，上式就化简为

$$(\hat{y}_{1k}+\hat{y}_{2k}+\cdots+\hat{y}_{nk})V_k=0$$

因为

$$V_k\neq 0$$

则有

$$(\hat{y}_{1k}+\hat{y}_{2k}+\cdots+\hat{y}_{nk})=0$$

所以，不定导纳矩阵每列元素之和等于零。

我们也可对$\hat{Y}$的各行来证明类似的结果，即$\hat{Y}$任一行元素之和为零。

因此，我们有如下结论。

定理13.4 电网络的不定导纳矩阵是一个等余子式矩阵。 □

不定导纳矩阵的另一个引人注目的性质就是：不定导纳矩阵是网络的一种非常基本的表达方式，在某种意义上讲，网络的多种描述都可从不定导纳矩阵得到，例如，划去不定导纳矩阵的i行与i列，即可得到以i端子为参考点的节点导纳矩阵。至于不定导纳矩阵在有源网络分析中应用的详细讨论，读者可参阅文献[13.3]。

下面研究不定导纳矩阵余子式的拓扑计算方法。

给定具有n个节点的网络N的不定导纳矩阵Y，首先按下列步骤构造一个加权有向图$G(\hat{Y})$。

1. $G(\hat{Y})$具有n个节点，用1、2、…n表示之。

2. 如果$\hat{y}_{ij}\neq 0$（i, $j=1$, 2, …n, $i\neq j$），那么在$G(\hat{Y})$中就存在一条从节点i指向节点j的边，其相应的权为$-y_{ij}$。

例如，图13.6(a)画出了对应于式（13.23）的不定导纳矩阵的有向图。

下面将用T_i来表示$G(\hat{Y})$中以节点i为根的一个有向生成树。$G(\hat{Y})$中的一个有向生成2-树是$G(Y)$的基本无向图中Y的一个生成2-树，并满足该有向2-树的每一个连通片都是$G(\hat{Y})$的一个有向树。我们也用$T_{ij,kl}$来表示满足下列条件的一个有向生成2-树：

1. 节点i和j在一片上，而节点k和l在另一片上；

2. 节点i和k是$T_{ij,kl}$的两片的根。

用类似的方法，可以定义有向生成3-树$T_{ij,kl,rs}$。节点i、k、r分别是$T_{ij,kl,rs}$三个片的根。有向生成树、有向生成2-树、有向生成3-树，如图13.6(b)所示。

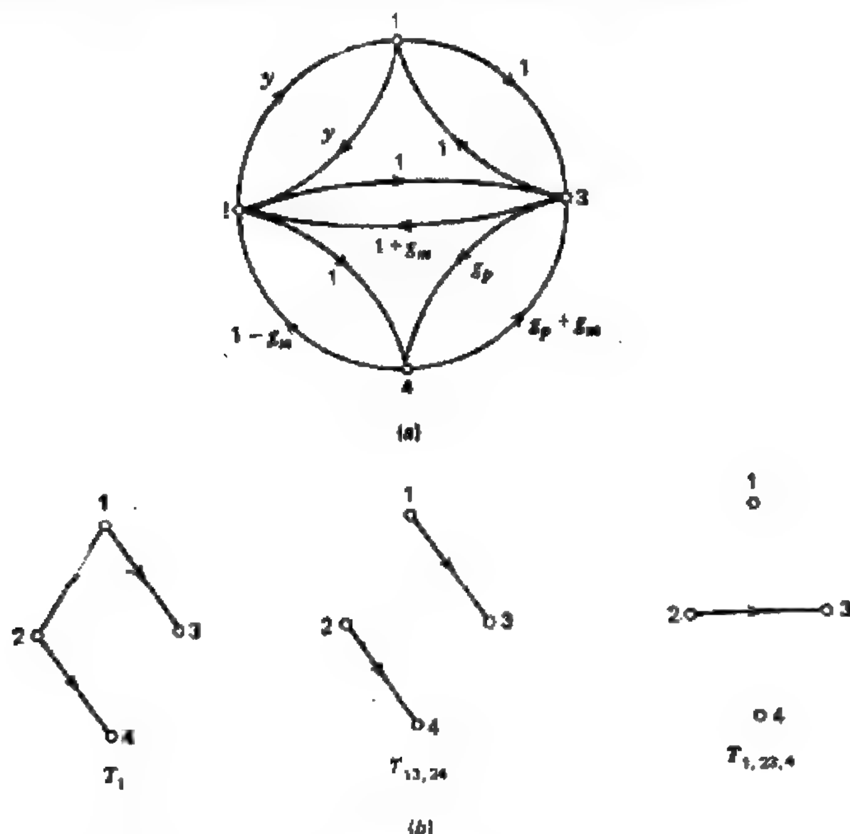


图 13.6(a)对应于式(13.23)不定导纳矩阵的图；(b)有向树，有向2-树，有向3-树。

我们再来定义：

$W_i=G(\hat{Y})$的全部有向生成树T_i的导纳积之和

$W_{ij,kl}=G(\hat{Y})$的全部有向生成2-树$T_{ij,kl}$的导纳积之和

$W_{ij,kl,rs}=G(\hat{Y})$的全部有生成3-树$T_{ij,kl,rs}$的导纳积之和

可以证明（习题6.18），

$$\hat{\Delta}_{kk}=W_k \qquad (13.24)$$

下述定理将$\hat{Y}$的二阶余子式与相应的有向2-树的导纳积联系起来。

定理13.5 设$\hat{Y}$表示网络N的不定导纳矩阵。那么

$$\hat{\Delta}_{ij,kk}=W_{ij,k}$$

证明

设G^*是把$G(\hat{Y})$的边（i, j）、（k, j）、（i, k）的权$-y_{ij}$、$-y_{kj}$、$-y_{ik}$分别改为$-y_{ij}+K$、$-y_{kj}-K$、$-y_{ik}-K$后而形成的有向图，$\hat{Y}^*$是相应的等余子式矩阵。我们可将G^*中的（i, j）边视为分别由具有权$-y_{ij}$和K的两条平行边（i, j）$_1$和$(i, j)_2$组成。类似地，（k, j）边也可视为分别由具有权$-y_{kj}$和$-K$的两条平行边$(k, j)_1$和$(k,j)_2$组成。

很显然，

$$\hat{\Delta}^*_{kk}=W^*_k \qquad (13.25)$$

这里"$*$"代表与$\hat{Y}^*$和G^*有关的量。

把G^*全部有向生成树T^*_k的集合S分为三个如下定义的集合S_1、S_2、S_3：

$S_1=\{T^*_k|T^*_k\in S$, 且$(i, j)_2$和$(k, j)_2$均不在T^*_k中$\}$

$S_2=\{T^*_k|T^*_k\in S$, $(k,j)_2$含于T^*_k, 并在T^*_k中存在一条从j到i的有向路径$\}$

$S_3=S-(S_1\cup S_2)$

既然在G^*中令$K=0$，就可得到G，那么，

$$\hat{\Delta}_{kk}=\sum_{T^*_k\in S_1}T^*_k\text{的导纳积} \qquad (13.26)$$

对于任何有向生成树$T^*_k\in S_2$, $T^*_k-(k, j)_2$都是$G(\hat{Y})$的一个有向生成2-树$T_{ij,k}$，反之亦然。因此

$$\sum_{T^*_k\in S_2}T^*_k\text{的导纳积}=-K\sum_{T_{ij,k}}T_{ij,k}\text{的导纳积}=-KW_{ij,k} \qquad (13.27)$$

注意：S_3中任一有向树T^*_k均为如下类型之一：

1.T^*_k包含$(i, j)_2$。

2.T^*_k包含$(k, j)_2$，且T^*_k中不存在从j到i的有向路径。

显而易见，对于每个前一类型的$T^*_k\in S_3$，存在唯一的后一类型的有向树$\{T^*_k-(i, j)_2\cup(k, j)_2\}\in S_3$，反之亦然。并且其中之一的导纳积是另一的负值。所以，

$$\sum_{T^*_k\in S_3}\text{有向树}T^*_k\text{的导纳积}=0 \qquad (13.28)$$

由式（13.26）到（13.28），可得

$$\hat{\Delta}^*_{kk}=\hat{\Delta}_{kk}-KW_{ij,k} \qquad (13.29)$$

最后，我们也具有如下关系：

$$\hat{\Delta}^*_{kk}=\hat{\Delta}_{kk}-K\hat{\Delta}_{ij,kk} \qquad (13.30)$$

根据式（13.29）和（13.30），则有

$$\hat{\Delta}_{ij,kk}=W_{ij,kk}$$ □

运用定理13.5，易于证明

$$\hat{\Delta}_{ii,jj,kk}=W_{i,j,k} \tag{13.31}$$

现在我们继续推导一般集中参数线性网络开路及短路函数的拓扑公式。

参看图13.7的双口网络N，设Y表示以n为参考节点的节点导纳矩阵，该矩阵可通过在N的不定导纳矩阵$\hat{Y}$中划去第n行和第n列而得到，即

$$Y=\hat{Y}_{nn} \tag{13.32}$$



图 13.7 双口网络

根据节点方程

$$YV=I$$

N的开路阻抗和短路导纳矩阵分别为

$$Z_{oc}=\frac{1}{\Delta}\begin{bmatrix}\Delta_{11} & \Delta_{21}-\Delta_{31}\\ \Delta_{12}-\Delta_{13} & \Delta_{22}+\Delta_{33}-\Delta_{23}-\Delta_{32}\end{bmatrix} \tag{13.33}$$

$$Y_{sc}=\frac{1}{\Delta_{11,22}+\Delta_{11,33}-\Delta_{11,23}-\Delta_{11,32}}\begin{bmatrix}\Delta_{22}+\Delta_{33}-\Delta_{23}-\Delta_{32} & \Delta_{31}-\Delta_{21}\\ \Delta_{13}-\Delta_{12} & \Delta_{11}\end{bmatrix} \tag{13.34}$$

为了求出Z_{oc}和Y_{sc}元素的拓扑公式，运用式（13.24），经观察得

$$\Delta=\hat{\Delta}_{nn}=W_{n}$$

再根据定理13.5，则有

$$\Delta_{11}=\hat{\Delta}_{11,nn}=W_{1,n} \tag{13.35}$$

$$\begin{aligned}\Delta_{21}-\Delta_{31}&=\hat{\Delta}_{21,nn}-\hat{\Delta}_{31,nn}\\&=W_{12,n}-W_{13,n}\\&=W_{123,n}+W_{12,n3}-W_{132,n}-W_{13,n2}\\&=W_{12,n3}-W_{13,n2}\end{aligned} \tag{13.36}$$

$$\Delta_{12}-\Delta_{13}=W_{21,3n}-W_{31,2n} \tag{13.37}$$

至此

$$\begin{aligned}\Delta_{22}+\Delta_{33}-\Delta_{32}-\Delta_{23}&=\hat{\Delta}_{22,nn}+\hat{\Delta}_{33,nn}-\hat{\Delta}_{32,nn}-\hat{\Delta}_{23,nn}\\&=\hat{\Delta}_{22,33}\end{aligned} \tag{13.38}$$

上述推导的最后一步是根据如下恒等式（参见文献[13.4]）：

$$\hat{\Delta}_{pq,rs}=\hat{\Delta}_{pq,uv}+\hat{\Delta}_{rs,uv}-\hat{\Delta}_{ps,uv}-\hat{\Delta}_{rq,uv}$$

所以，根据定理13.5，则得

$$\Delta_{22}+\Delta_{33}-\Delta_{32}-\Delta_{23}=\hat{\Delta}_{22,33}=W_{2,3} \tag{13.39}$$

进而

$$\begin{aligned}
&\Delta_{11,22}+\Delta_{11,33}-\Delta_{11,23}-\Delta_{11,32}\\
&=\hat{\Delta}_{11,22,nn}+\hat{\Delta}_{11,33,nn}-\hat{\Delta}_{11,23,nn}-\hat{\Delta}_{11,32,nn}\\
&=W_{1,2,n}+W_{1,3,n}-W_{1,32,n}-W_{1,23,n}\\
&-W_{13,2,n}+W_{1,2,n3}+W_{12,3,n}+W_{1,3,n2}
\end{aligned} \tag{13.40}$$

根据上面的方程，有如下定理。

定理13.6 设N是图13.7所示的具有正、负参考端子的双口网络，那么N的开路阻抗矩阵和短路导纳矩阵为

$$Z_{oc}=\frac{1}{W_n}\begin{bmatrix} W_{1,n} & W_{12,n3}-W_{13,n2} \\ W_{21,3n}-W_{31,2n} & W_{2,n} \end{bmatrix}$$

$$Y_{sc}=\frac{1}{W_{13,2,n}+W_{1,2,n3}+W_{13,3,n}+W_{12,n3}}\begin{bmatrix} W_{2,3} & W_{13,n}-W_{12,n3} \\ W_{31,2n}-W_{21,3n} & W_{1,n} \end{bmatrix}$$

□

下面我们举例说明采用定理13.6来求Z_{oc}和Y_{sc}的拓扑计算方法。

再考虑图13.5中的网络N，其中节点1和4形成1口，节点2和4形成2口。N的不定导纳矩阵$\hat{Y}$已由式(13·23)给出，对应于$\hat{Y}$的图$G(\hat{Y})$，如图13.6(a)所示。

为了采用定理13.6来计算Z_{oc}和Y_{sc}，就需求出$G(\hat{Y})$的W_4、$W_{1,4}$、$W_{2,4}$、$W_{21,4}$和$W_{1,2,4}$。首先找出所需的$G(\hat{Y})$的生成树、生成2-树和生成3-树：

T_4

{(2,1), (4,2), (4,3)}, {(4,2), (4,3), (3,1)},
{(1,2), (3,1), (4,3)}, {(2,1), (4,2), (1,3)},
{(2,1), (2,3), (4,2)}, {(3,1), (3,2), (4,3)},
{(2,1), (3,2), (4,3)}, {(3,1), (2,3), (4,2)}

（注意：(i,j)是从i指向j的边）。

$T_{1,4}$

{(1,2), (4,3)}, {(1,2), (1,3)}, {(1,2), (2,3)}, {(1,3), (3,2)},
{(1,3), (4,2)}, {(3,2), (4,3)}, {(4,2), (2,3)}, {(4,2), (4,3)}

$T_{2,4}$

{(2,1), (4,3), {(2,1), (1,3)}, {(2,1), (2,3)},
{(3,1), (2,3)}, {(3,1), (4,3)}

$T_{12,4}$

$\{(1,2),\ (1,3)\},\ \{(1,2),\ (2,3)\},\ \{(1,3),\ (3,2)\},\ \{(1,2),\ (4,3)\}$

$T_{21,4}$:

$\{(2,1),\ (1,3)\},\ \{(2,1),\ (2,3)\},\ \{(3,1),\ (2,3)\},\ \{(2,1),\ (4,3)\}$

$T_{1,2,4}$:

$$\{(1,3)\},\ \{(2,3)\},\ \{(4,3)\}$$

运用上式可得如下导纳积：

$$W_4=g_m(y+1)+g_p(3y+2)+(2y+1)$$
$$W_{1,4}=(2+y)(2+g_p+g_m)-(1+g_m)$$
$$W_{2,4}=(1+y)(2+g_p+g_m)-1$$
$$W_{12,4}=y(2+g_p+g_m)+(1+g_m)$$
$$W_{21,4}=y(2+g_p+g_m)+1$$
$$W_{1,2,4}=2+g_p+g_m$$

因而得到

$$Z_{oc}=\frac{1}{W_4}\begin{bmatrix} W_{1,4} & W_{12,4} \\ W_{21,4} & W_{2,4} \end{bmatrix}$$

$$=\frac{1}{g_m(y+1)+g_p(3y+2)+(2y+1)}$$

$$\times\begin{bmatrix} (2+y)(2+g_p+g_m)-(1+g_m) & y(2+g_p+g_m)+(1+g_m) \\ y(2+g_p+g_m)+1 & (1+y)(2+g_p+g_m)-1 \end{bmatrix}$$

$$Y_{sc}=\frac{1}{W_{1,2,4}}\begin{bmatrix} W_{2,4} & -W_{12,4} \\ -W_{21,4} & W_{1,4} \end{bmatrix}$$

$$=\frac{1}{2+g_p+g_m}\begin{bmatrix} (1+y)(2+g_p+g_m)-1 & -y(2+g_p+g_m)-(1+g_m) \\ -y(2+g_p+g_m)-1 & (2+y)(2+g_p+g_m)-(1+g_m) \end{bmatrix}$$

13.3 伴随网络和网络灵敏度的计算

本节介绍一个计算网络灵敏度的方法，它是根据特勒根定理和伴随网络的概念提出的（参见文献[13.5]和[13.6]）。下面的论述方法基于文献[13.6]。

设N是一个集中参数线性时不变网络，它含有电阻、电容、电感、变压器、回转器、电压控制电压和电流源、电流控制电压和电流源。下面的讨论中，尽管对于多于双口的N成立，仍假定N是一个双口网络。

令$\tilde{N}$是与N拓扑意义等价的双口网络。换言之，$\tilde{N}$的图和N的图是等同的。N与$\tilde{N}$的对应元件都用相同的符号表示。注意：此时我们尚未定义$\tilde{N}$的元件及其数值。

设V_e、I_e分别表示N中某元件e的电压和电流；ψ_e、λ_e分别表示$\tilde{N}$中对应于元件e的的电压和电流。V_i、I_i（$i=1,2$）表示N中端口的电压、电流变量，而ψ_i、λ_i（$i=1,2$）表示$\tilde{N}$中端口的相应变量。

对于网络N与$\widetilde{N}$，应用特勒根定理，则有

$$V_1\lambda_1+V_2\lambda_2=\sum_e V_e\lambda_e \qquad (13.41a)$$

和

$$I_1\psi_1+I_2\psi_2=\sum_e I_e\Psi_e \qquad (13.41b)$$

这里求和是对N与$\widetilde{N}$的全部元件进行的．假定网络N的元件值发生变化，那么对该扰动网络N和网络$\widetilde{N}$应用特勒根定理，可得

$$(V_1+\Delta V_1)\lambda_1+(V_2+\Delta V_2)\lambda_2=\sum_e(V_e+\Delta V_e)\lambda_e \qquad (13.42a)$$

$$(I_1+\Delta I_1)\psi_1+(I_2+\Delta I_2)\psi_2=\sum_e(I_e+\Delta I_e)\psi_e \qquad (13.42b)$$

这里ΔV和ΔI表示N中由于元件值的变化所引起电压、电流的波动。

从式(13,42)减去式(13,41)，则有

$$\Delta V_1\lambda_1+\Delta V_2\lambda_2=\sum_e\Delta V_e\lambda_e \qquad (13.34)$$

和

$$\Delta I_1\psi_1+\Delta I_2\psi_2=\sum_e\Delta I_e\psi_e \qquad (13.44)$$

再从式(13,43)减去式(13,44)，则有

$$(V_1\lambda_1-\Delta I_1\psi_1)+(\Delta V_2\lambda_2-\Delta I_2\psi_2)=\sum_e(\Delta V_e\lambda_e-\Delta I_e\psi_e) \qquad (13.45)$$

现在我们希望定义$\widetilde{N}$中元件，并使得式(13.45)与所有ΔV_e和ΔI_e项无关。

先考虑电阻元件．我们有

$$V_R=RI_R \qquad (13,46)$$

这里V_R和I_R是电阻元件R上的电压和电流．设R变为$R+\Delta R$．那么

$$V_R+\Delta V_R=(R+\Delta R)(I_R+\Delta I_R)$$

忽略式中二次项，并化简为

$$V_R+\Delta V_R=RI_R+R\Delta I_R+I_R\Delta R \qquad (13.47)$$

从式（13.47）减去式（13.46），则得

$$\Delta V_R=R\Delta I_R+I_R\Delta R \qquad (13.48)$$

因此，式（13.45）中对应于N的电阻元件的各项可写为

$$\sum_R[(R\lambda_R-\psi_R)\Delta I_R+I_R\lambda_R\Delta R] \qquad (13.49)$$

这里求和是对N中全部电阻元件进行的。注意：λ和ψ的下标并非指明$\widetilde{N}$中的元件一定是电阻类型，而只是指明N与$\widetilde{N}$元件之间的对应性。

如果选择

$$\psi_R=R\lambda_R \qquad (13.50)$$

式（13.49）就简化为

$$\sum_R I_R\lambda_R\Delta R \qquad (13.51)$$

此式与ΔV_R和ΔI_R无关。

因为方程（13.50）是值为R的电阻关系式，所以$\widetilde{N}$中对应N的、值为R的电阻元件也

是一个值为R的电阻。

再考虑下述关系定义的电压控制电压源：

$$V_{VV_2}=\mu V_{VV_1}$$

和

$$I_{VV_1}=0$$

因此，忽略式中二次项，则得

$$\Delta V_{VV_2}=\mu\Delta V_{VV_1}+V_{VV_1}\Delta\mu$$

和

$$\Delta I_{VV_1}=0$$

式（13.45）中对应于电压控制电压源的各项现在可写为

$$\sum[(\lambda_{VV_1}+\mu\lambda_{VV_2})\Delta V_{VV_1}-\psi_{VV_2}\Delta I_{VV_2}+V_{VV_1}\lambda_{VV_2}\Delta\mu] \tag{13.52}$$

如若选择

$$\lambda_{VV_1}=-\mu\lambda_{VV_2}$$
$$\psi_{VV_2}=0 \tag{13.53}$$

那么式（13.52）就化简为

$$\sum V_{VV_1}\lambda_{VV_2}\Delta\mu$$

注意：式（13.53）描述了一个放大倍数为$-\mu$的电流控制电流源，并且在伴随网络中的施控与受控元件之间的地位变化。

在式（13.45）里，网络N中其余元件类型所对应的各项也可用类似的方法得到。表13.1给出了对于相关网络$\widetilde{N}$来选择适当支路关系的结果。

表 13.1

元件类型	支路关系	伴随网络中支路关系	灵敏度 （ψ的组成部分）	Δp的组成部分
电阻	$V_R=RI_R$	$\psi_R=R\lambda_R$	$+I_R\lambda_R$	ΔR
电导	$I_G=GV_G$	$\lambda_G=G\psi_G$	$-V_G\psi_G$	ΔG
电容	$I_C=j\omega CV_C$	$\lambda_C=j\omega C\psi_C$	$-j\omega V_C\psi_C$	ΔC
倒电容	$SI_C=j\omega V_S$	$S\lambda_S=j\omega\psi_S$	$+\frac{1}{j\omega}I_S\lambda_S$	ΔS
电感	$V_L=j\omega LI_L$	$\psi_L=j\omega L\lambda_L$	$+j\omega I_L\lambda_L$	ΔL
倒电感	$\Gamma V_\Gamma=j\omega I_\Gamma$	$\Gamma\psi_\Gamma=j\omega\lambda_\Gamma$	$-\frac{1}{j\omega}V_\Gamma\psi_\Gamma$	$\Delta\Gamma$
变压器	$V_{T_1}=nV_{T_2}$ $I_{T_2}=-nI_{T_1}$	$\psi_{T_2}=n\psi_{T_1}$ $\lambda_{T_1}=-n\lambda_{T_2}$	$+(V_{T_2}\lambda_{T_1}+I_{T_1}\psi_{T_2})$	Δn
回转器	$V_{GY_1}=\alpha I_{GY_2}$ $V_{GY_2}=-\alpha I_{GY_1}$	$\psi_{GY_1}=-\alpha\lambda_{GY_2}$ $\psi_{GY_2}=\alpha\lambda_{GY_1}$	$+(I_{GY_2}\lambda_{GY_1}-I_{GY_1}\lambda_{GY_2})$	$\Delta\alpha$
电压控制 电压源	$V_{VV_2}=\mu V_{VV_1}$ $I_{VV_1}=0$	$\lambda_{VV_1}=-\mu\lambda_{VV_2}$ $\psi_{VV_2}=0$	$+V_{VV_1}\lambda_{VV_2}$	$\Delta\mu$
电压控制 电流源	$I_{VI_2}=g_mV_{VI_1}$ $I_{VI_1}=0$	$\lambda_{VI_1}=g_m\psi_{VI_2}$ $\lambda_{VI_2}=0$	$-V_{VI_1}\psi_{VI_2}$	Δg_m
电流控制 电压源	$V_{IV_2}=r_mI_{IV_1}$ $V_{IV_1}=0$	$\psi_{IV_1}=r_m\lambda_{IV_2}$ $\psi_{IV_2}=0$	$+I_{IV_1}\lambda_{IV_2}$	Δr_m
电流控制 电流源	$I_{II_2}=\beta I_{II_1}$ $V_{II_1}=0$	$\psi_{II_1}=-\beta\psi_{II_2}$ $\lambda_{II_2}=0$	$-I_{II_1}\psi_{II_2}$	$\Delta\beta$

如果$\widetilde{N}$的元件均选表13.1给出的那样，那么式（13.45）就化简为

$$\sum(\Delta V_1\lambda_1-\Delta I_1\psi_1)+(\Delta V_2\lambda_2-\Delta I_2\psi_2)=g^t\Delta p \tag{13.54}$$

这里g和Δp都是向量，它们的元素由表13.1给定。

元件由表13.1所定义的网络$\widetilde{N}$称为N的伴随。在图13.8里，列出了N的各种网络元件及其在伴随网络$\widetilde{N}$中所对应的元件。

注意，伴随网络$\widetilde{N}$与原网络N有如下关系：

1. $\widetilde{N}$的图与N的图相同。

2. N中所有的电阻、电容、电感元件分别对应于$\widetilde{N}$中具有相同值的电阻、电容、电感元件。

3. N中所有匝数比为$1:n$的变压器对应于$\widetilde{N}$中变比为$1:n$的变压器。

4. N中所有回转系数为α的回转器对应于$\widetilde{N}$中所有回转系数为$-\alpha$的回转器（或称回转器两门互换）。

5. N中具有电压放大倍数为μ的电压控制电压源对应于$\widetilde{N}$中具有电流放大倍数为$-\mu$的电流控制电流源，并且N中施控与受控元件的地位在$\widetilde{N}$中是相反的。

6. N中具有电流放大倍数为β的电流控制电流源对应于$\widetilde{N}$中具有电压放大倍数为$-\beta$的电压控制电压源，并且N中施控与受控元件的地位在$\widetilde{N}$中是相反的。

7. N中电压控制电流源和电流控制电压源对应于$\widetilde{N}$中的电压控制电流源和电流控制电压源，并且N中施控与受控元件的地位在$\widetilde{N}$中是相反的。

现在，我们来说明伴随网络在计算网络函数灵敏度时的应用。

对于某网络中某一参数x的网络函数$F(s)$，其灵敏度S_x^F定义为

$$S_x^F=\frac{x}{F}\frac{\partial F}{\partial x}$$

很显然，灵敏度S_x^F是x的增量对$F(s)$影响的量度。计算S_x^F主要取决于求解$\frac{\partial F}{\partial x}$，计算$S_x^F$可按下述方法进行：

1. 选择端口变量，并使式（13.54）左端化简为ΔF（即在网络N和伴随网络$\widetilde{N}$的端口上加入适当的激励）。比如，$F(s)$是开路电压比，即

$$F(s)=\frac{V_2(s)}{V_1(s)}$$

那么网络N与$\widetilde{N}$的激励选择如下：

a. 在N的端口1处跨接一个值为常数1的独立电压源，则有

$$V_1=1,\ \Delta V_1=0$$

b. 将N的端口2开路，则有

$$I_2=0$$

c. 将$\widetilde{N}$的端口1短路，则有

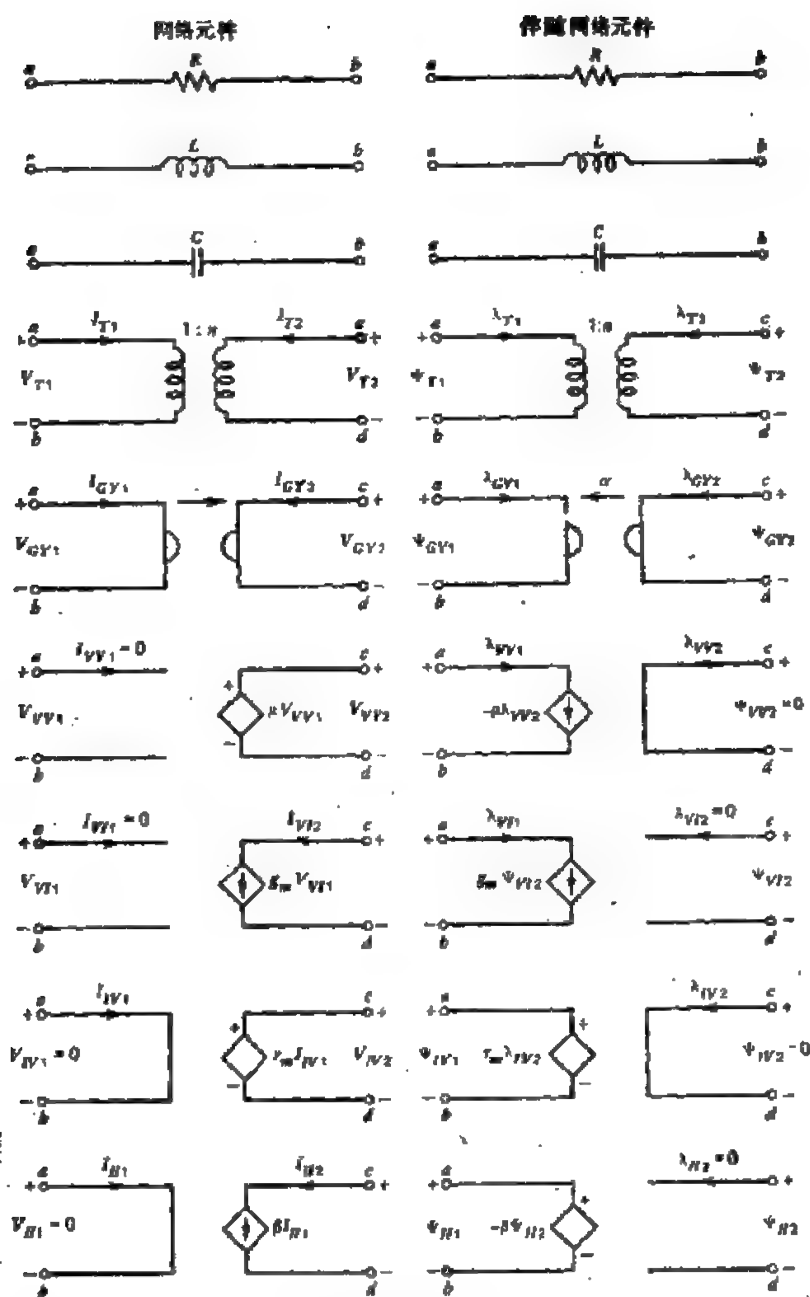


图 13.8 网络元件及其伴随网络元件

$$\psi_1 = 0$$

d. 在$\widetilde{N}$的端口2处跨接一个值为 1 的独立电流源，则有

$$\lambda_2 = 1$$

我们很容易证明，以上选择的端口激励可将式（13.54）左端化简为

$$\Delta V_2 = \Delta F$$

2. 对于加了上述激励的 N 与 $\widetilde{N}$ 进行分析，并且求出全部元件的电压电流值，

3. 式（13.54）右端对应参数x的项，是与该参数有关的电压（或电流）同 Δx 之积

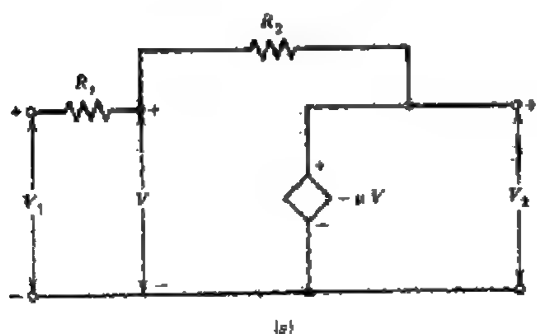
（见表13.1）。因此，只要N与$\widetilde{N}$的电压和电流是由2解出的，那么对于每个x就都可求解$\frac{\partial F}{\partial x}$。

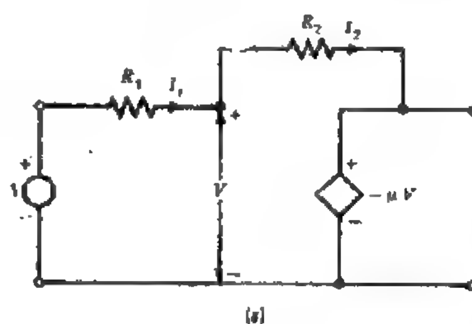
所以，用伴随网络的概念来进行灵敏度计算，就需要对给定网络N及其伴随网络$\widetilde{N}$进行分析。

现举例来说明上述灵敏度计算方法，该例选自文献[13.3]。

考虑图13.9(a)所示网络N，其伴随网络$\widetilde{N}$如图13.9(b)所示。N的开路电压比为

$$F(s)=\frac{V_2(s)}{V_1(s)}=\frac{-\mu R_2}{R_1+R_2+\mu R_1}$$





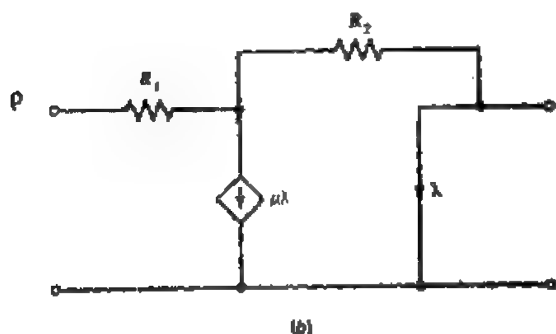


图 13.9 网络N及其伴随网络$\widetilde{N}$

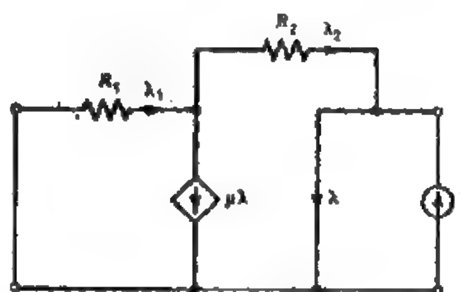


图 13.10 图13.9中网络的端口激励

按前面的步骤，给N与$\widetilde{N}$加上图13.10所示的激励。由表13.1得

$$\frac{\partial F}{\partial(-\mu)}=V\lambda$$

$$\frac{\partial F}{\partial R_1}=I_1\lambda_1$$

$$\frac{\partial F}{\partial R_2}=I_2\lambda_2$$

很容易证明

$$I_1=I_2=\frac{1+\mu}{R_1+R_2+\mu R_1}$$

$$V=\frac{R_2}{R_1+R_2+\mu R_1}$$

$$\lambda=\frac{R_1+R_2}{\mu R_1+R_1+R_2}$$

$$\lambda_1=\lambda+\mu\lambda+1=\frac{\mu R_2}{\mu R_1+R_1+R_2}$$

$$\lambda_2-\lambda-1=\frac{-\mu R}{\mu R_1+R_1+R_2}$$

所以，

$$S_{\mu}^{F}=\frac{R_1+R_2}{R_1+R_2+\mu R_1}$$

$$S_{R_1}^{F}=\frac{-R_1(\mu+1)}{R_1+R_2+\mu R_1}$$

$$S_{R_2}^{F}=\frac{R_1(1+\mu)}{R_1+R_2+\mu R_1}$$

在第十一章(习题11.3)中我们知道，一个平面网络的回路阻抗矩阵等于其对偶网络的割集导纳矩阵。现在，我们要提出这样的问题：对于一个给定的网络N，是否存在另一个N_T，从而使N_T的不定导纳矩阵等于N的不定导纳矩阵的转置？答案是肯定的。文献[13.7]已经证明如何来构造一个称为N的转置的网络N_T，恰好N_T正和N的伴随网络是一样的。参见文献[13.8]。

文献[13.9]证明了一个与特勒根定理相类似的定理，但是这个定理可用于任何两个具有对偶拓扑性质的平面网络。运用该定理，可以从一给定的平面网络得到一个称为广义的对偶转置的新网络。如果N_D是N的对偶，并且N_{DT}是N_D的转置，那么可以证明，N_{DT}是N的广义对偶转置的特殊情况。转置与广义对偶转置的概念已使我们发现了各种网络实现之间的统一性（在论文中有所论述），这些貌似不同的实现经转置和广义对偶转置处理后，实际上都是有内在联系的。

比如，文献[13.3]提出了实现电压传递函数的两种结构，并证明了其中任一种都是另一种的广义对偶转置。又如，文献[13.10]分别构造了电压传递函数的实现和电流传递函数的实现，其中，一种为另一种的转置。读者尚可在文献[13.9]中找到更多的例子，它们说明了怎样运用转置和广义对偶转置的概念，获得实现给定网络函数的不同等价结构。

13.4 进一步阅读

需要进一步了解关于详细讨论网络函数拓扑公式的读者，可首先参阅文献[13.11]。讨论拓扑公式的其它书籍，还有文献[13.12]、[13.13]和[13.14]。

本章所给出的一般网络的拓扑公式是根据对应于网络不定导纳矩阵的树而得到的。许多人已提出了直接由给定网络写出这些公式的方法，例如，文献[13.15]中就有大量的有关论文列为参考。用拓扑公式导出RLC网络的网络函数的许多有趣性质，请考阅文献[13.12]。

文献[13.16]、[13.17]和[13.18]给出了特勒根定理的一般形式及其应用。文献[13.19]叙述了如何在n口电阻网络的计算机辅助设计中，运用伴随网络的概念。在用网络函数来确定灵敏度不变性及元件灵敏度数值的总和的界限方面，也已开始引用了伴随网络的概念，可参阅文献[13.20]、[13.21]。

13.5 习　题

13.1 （a）证明：无互感RLC网络的导纳矩阵在$s=0$上有一个极点，当且仅当在端口的两个端点之间存在一条全电感路径。

（b）证明：无互感RLC网络的导纳矩阵在无穷远处有一个极点，当且仅当在端口的两个端点之间有一条全电容路径。

13.2 令

$$z_{11}=\frac{a_0+a_1s+\cdots+a_ns^n}{D(S)}$$

$$z_{22}=\frac{b_0+b_1s+\cdots b_ms^m}{D(S)}$$

$$z_{21}=\frac{c_0+c_1s+\cdots c_rs^r}{D(S)}$$

是无互感RLC二端口网络的开路阻抗函数。用拓扑公式证明：

$$a_k\geqslant|c_k|$$

和

$$b_k\geqslant|c_k|$$

13.3 推导出一般线性网络开路电压传输比和短路电流传输比的拓扑公式。

13.4 求图出13.11所示网络的策动点阻抗对应于R的灵敏度。

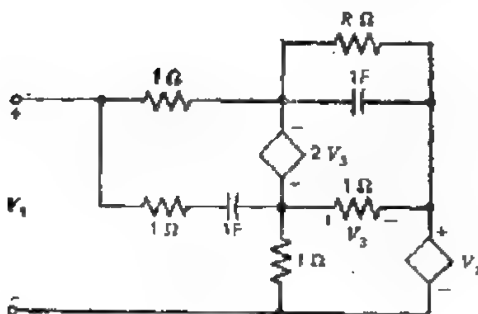


图 13.11

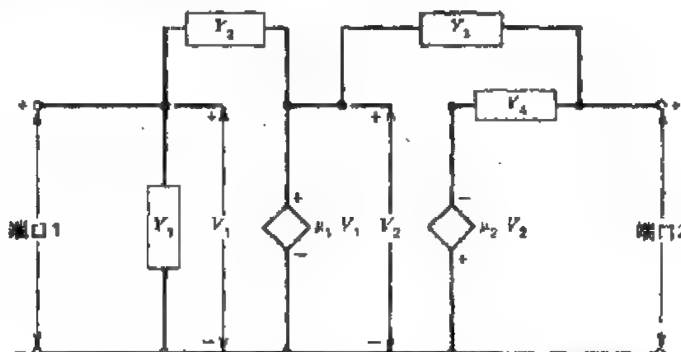


图 13.12

13.5 证明：网络N的不定导纳矩阵是它的伴随网络的不定导纳矩阵的转置。

提示：应用特勒根定理。

13.6 在表示含有多端口元件网络的图中，每个端口用一条边来表示。N和N_D是两个具有对偶图的平面网络。推导出一个类似特勒根定理的定理，此定理仅含有N和N_D的电流变量或电压变量。

13.7 设N是平面网络，N_D是N的对偶，N_{DT}是N_D的转置或伴随网络。利用习题13.6的定理，按照类似于13.3节所叙述的程序，直接从N（不构造N_D）得出网络N_{DT}的元件类型和数值（文献[13，9]）。

13.8 对于图13.12所示网络N，构造出习题13.7中定义的网络N_{DT}（N的转置对偶网络）。

13.6 参 考 文 献

13.1 A. C.Aitken, *Determinants and Matrices*, Interscience, New York, 1956.

13.2 N. Balabanian and T. A. Bickart, *Electrical Network Theory*, Wiley, New York, 1969.

13.3 S. K. Mitra, *Analysis and Synthesis of Linear Active Networks,* Wiley, New York, 1969.

13.4 G. E. Sharpe and B. Spain, "On the Solution of Networks by means of the Equi-Cofactor Matrix," *IRE Trans. Circuit Theory,* Vol. CT-7, 230—239(1960).

13.5 J. L. Bordewijk, "Inter-Reciprocity Applied to Electrical Networks," *Appl. Sci. Res.,* Vol. B6, 1—74(1956).

13.6 S. W. Director and R. A. Rohrer, "Automated Network Design-The Frequency Domain Case," *IEEE Trans. Circuit Theory,* Vol. CT-16, 330—337(1969).

13.7 B. B. Bhattacharyya and M. N. S. Swamy, "Network Transposition and Its Application in Synthesis," *IEEE Trans. Circuit Theory,* Vol. CT-18,394—397(1971).

13.8 M. N. S. Swamy, C. Bhushan, and B. B. Bhattacharyya, "Generalized Duals, Generalized Inverses, and Their Applications," *Radio and Electron. Eng.,* Vol. 44, 95—97(1974).

13.9 M. N. S. Swamy, C. Bhushan, and B. B. Bhattacharyya, "Generalized Dual Transposition and Its Applications," *J. Franklin Inst.,* Vol. 301, 465—474(1976).

13.10 K. Dagget and J. Vlach, "Sensitivity-Compensated Active Networks," *IEEE Trans. Circuit Theory,* Vol. CT-16, 416—422(1969).

13.11 W. K. Chen, *Applied Graph Theory,* North-Holland, Amsterdam,1971.

13.12 S. Seshu and M. B. Reed, *Linear Graphs and Electrical Networks,* Addison-Wesley, Reading, Mass., 1961.

13.13 S. P. Chan, *Introductory Topological Analysis of Electrical Networks,* Holt, Rinehart and Winston, New York,1969.

13.14 W. Mayeda, *Graph Theory,* Wiley-Interscience, New York, 1972.

13.15 J. Numata and M. Iri, "Mixed-Type Topological Formulas for General Linear Networks,"*IEEE Trans. Circuit Theory,* Vol. CT-20, 488—494(1973).

13.16 P. Penfield, Jr., R. Spence, and S. Duinker, *Tellegen's Theorem and Electrical Networks,* M. I. T Press, Cambridge, Mass., 1970.

13.17 P. Penfield, Jr., R. Spence, and S. Duinker, "A Generalized Form of Tellegen's Theorem," *IEEE Trans. Circuit Theory,* Vol. CT-17, 302—305(1970).

13.18 C. A. Desoer, "On the Description of Adjoint Networks," *IEEE Trans. Circuits and Sys.,* Vol. CAS-22, 585—587(1975), Erratum, *IEEE Trans. Circuits and Sys.,* Vol. CAS--23,58(1976).

13.19 S. W. Director and R. A. Rohrer, "On the Design of Resistance n-Port Networks by Digital Computer," *IEEE Trans. Circuit Theory,* Vol. CT-16, 337—346 (1969).

13.20 M. N. S. Swamy, C. Bhushan, and K. Thulasiraman, "Bounds on the Sum of Element Sensitivity Magnitudes for Network Functions," *IEEE Trans. Circuit Theory,* Vol CT-19, 502—504(1972).

13.21 M. N. S. Swamy, C. Bhushan, and K. Thulasiraman, "Sensitivity Invariants for Linter Time-Invariant Networks," *IEEE Trans. Circuit Theory,* Vol. CT-20, 21—24(1973).

第三篇　算法图论

第十四章　算法分析

图产生于若干实际问题的研究中。这些研究的第一步就是将所考虑的问题中含有的图论性质发现出来，这些性质将帮助我们提出解决问题的方法。例如，在第十一章到第十三章，我们建立了电网络的几个有用的图论性质，并且利用这些性质推导出了其中包括列出网络方程的各种方法。作为另一个例子，在研究运输网络时，我们常对确定一个最大流感兴趣。在第15.7节中，我们第一步首先将这些网络的几个特性逐步展开讨论，从而得出了“最大流最小切割定理”。这个定理为求最大流的 Ford-Fulkerson 标号算法奠定了基础。通常需要解决的问题包括了分析一个图，或针对某个特殊性质检查一个图。在实际问题研究中产生的图是非常大而又复杂的。因此，用有效地方法分析这类图就包括对有效的计算机算法的设计。

在本书的这一部分，我们将讨论一些图的算法，我们主要关心的是建立设计算法所依据的理论基础。我们也将推导出与某些算法的计算复杂性有关的结果。在某些情况下，计算复杂性决定性地依赖于实现某些基本运算的计算复杂性，如不相交集合的并（此例见第14.5节）。在这种情况下，我们对有兴趣进一步研究的读者提供了适当的参考文献。

一个算法的计算复杂性就是对该算法运行时间的测量，因此它是一个输入规模的函数。对图的算法来论，复杂性函数将以图的顶点数和边数来表示。其次，一个函数$g(n)$被认成是$O(f(n))$，当且仅当存在常数c和n_0，并对于所有的$n\geqslant n_0$，使得$|g(n)|\leqslant c\,|f(n)|$。此外，所有的复杂性函数都是在最坏情况下的分析得出的。

在计算机上存在多种表示图的方法，最常见的两种方法采用邻接矩阵法（第6.10节）与邻接表法。邻接矩阵表示法在稀疏图的情况下并不是十分有效的。在邻接表表示法中，我们把一张表与每个顶点联系起来，该表包含了与每个顶点相关联的所有边。表示一个图的数据结构的详细讨论，可参阅本章末所列的某些参考书。

有些算法虽然设计很困难，但其正确性的证明却很明显；而另外一些算法，虽然容易想象，但其正确性的证明却很困难。此外，有些很平凡的算法但进行分析却不容易。这些类型算法的实例可在本章以及下一章的各节中找到。

为在这里讨论起见，我们将算法分为两组：有关图的分析的算法和有关图的优化问题的算法。本章，我们将研究第一组算法。广义地说，本章将讨论与下面主题有关的算法：

1.传递闭包，

2.传递定向，

3.深度优先搜索，

4.2-连通性和强连通性，

5.程序图。

14.1 传递闭包

我们曾记得（第5.2节）：一个集合的二元关系就是该集合中元素的有序对的集成。一个二元关系R的传递闭包就是定义如下的关系R^*：

xR^*y，当且仅当存在一个序列

$$x_0=x,\ x_1,\ x_2,\ \cdots,\ x_k=y$$

并使得$k>0$且x_0Rx_1，x_1Rx_2，…，$x_{k-1}Rx_k$。

很清楚，如果xRy，那么xR^*y。因此，$R\subseteq R^*$。进而很容易证明：R^*是可传递的。事实上，R^*是包含R的最小传递关系。因此，如果R是可传递的，那么$R^*=R$。

正如第5.2节中已指出的那样，一个二元关系可以用一个有向图来表示。假设G是表示关系R的有向图。表示R的传递闭包R^*的有向图G^*称为G的传递闭包。由R^*的定义可知：边（x，y）在G^*中，$x\neq y$，当且仅当G中存在一条从顶点x到顶点y的有向路径。类似地，顶点x的自环（x，x）在G^*中，当且仅当G中存在一条包含x的有向回路。例如，图14.1(b)所示图就是14.1(a)所示图的传递闭包。

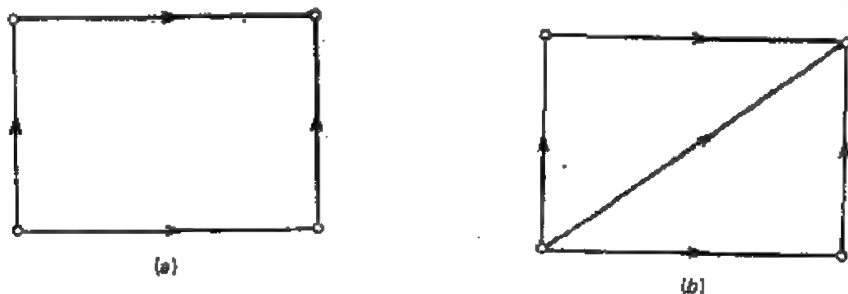


图 14.1 (a)图G；(b)图G^*，G的传递闭包。

假设，我们定义n-顶点有向图G的可达性矩阵为一个$n\times n$(0，1)矩阵，其中（i，j）元等于1，当且仅当$i\neq j$时存在一条从顶点i到顶点j的有向路；当$i=j$时，有一条含i的有向回路。换言之，可达性矩阵的（i，j）元等于1，当且仅当顶点j是从顶点i出发，经过一系列有向边可到达的。显而易见，G^*的邻接矩阵与G的可达性矩阵是相同的。

构造一个有向图的传递闭包问题是从一些应用中产生的。例如，见文献[14.1]。本节，我们将讨论一个优美的、并且非常有效的计算传递闭包的算法，这一算法是由Warshall在文献[14.2]中提出的。我们还讨论由Warren在文献[14.3]中给出的Warshall算法的一个变形。

令G为一个n-顶点有向图，其顶点用整数1，2，…，n表示。令$G^0=G$。Warshall算法构造了一个图序列，并使得$G^i\subseteq G^{i+1}$，$0\leqslant i\leqslant n-1$，且$G^n$是$G$的传递闭包。图$G^i$（$i\geqslant 1$）是在$G^{i-1}$中、通过对$G^{i-1}$中顶点$i$进行处理而得到的。在$G^{i-1}$中处理顶点$i$就包含了增加一条到$G^{i-1}$的新边，其定义如下：

在G^{i-1}中，令（i，k）、（i，l）、（i，m）、…是与顶点i相关联的射出边。那么对每一条与顶点i相关联的射入边（j，i），都增加边（j，k）、（j，l）、（j，m）、…到G^{i-1}上，如果这些边不在G^{i-1}中的话。对顶点i作处理后所产生的图就记为G^i。

Warshall算法示于图14.2。很显然，$G^i \subseteq G^{i+1}$，$i \geq 0$。为了证明G^n是G的传递闭包，我们需要证明下面的结论。

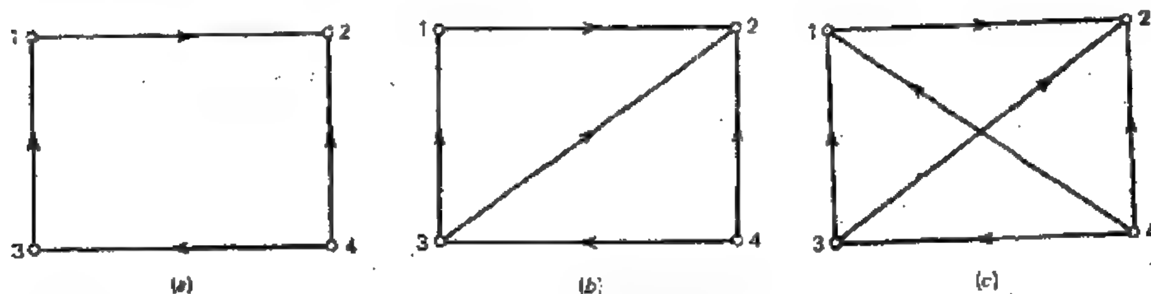


图 14.2 Warshal算法的举例说明
$(a)G^0$；$(b)G^1=G^2$；$(c)G^3=G^4$。

定理14.1

1. 假设：对于任意两个顶点s和t，G中都存在一条从顶点s到顶点t的有向路径P，并使得除s和t以外的所有顶点都在集合{1，2，…，i}中。那么G^i就包含了边（s，t）。

2. 假设：对于任意顶点s，则G中存在一条包含s的有向回路C，并使得除s以外的所有顶点都在集合{1，2，…i}中。那么G^i就包含了自环(s，s)。

证明

1. 对i采用归纳法来证明。

如果G^0（$=G$）包含边（s，1）和（1，t），结论显然对G^1为真，这是因为用Warshall的构造法在处理顶点1后引进了边（s，t）。

设对所有G^k，$k<i$，结论都为真。

假设i不是P的内部顶点。那么由归纳法假设G^{i-1}包含边（s，t）。因此G^i也包含（s，t），因为$G^{i-1} \subseteq G^i$。

假设i是P的内部顶点。那么，由归纳法假设可知：G^{i-1}包含边（s，i）和（i，t）。因此，在G^{i-1}中顶点i经处理后，边（s，t）就被加到G^i上。

2. 用上述方法可证明。 □

作为本定理的一个直接推论如下：

推论14.1.1 G^n是G的传递闭包。 □

下面给出Warshall算法的一个正式描述。在这个描述中，图G用其邻接矩阵M表示，符号$\vee$代表布尔和。

算法14.1 传递闭包（Warshall）

S1 （初始化）M为G的邻接矩阵。

S2 对于$i=1$，2，…，n，执行S3。

S3 对于$j=1$，2，…，n，执行S4。

S4 若$M(j, i)=1$，对于$k=1$，2，…，n，执行S5。

S5 $M(j, k)=M(j, k) \vee M(i, k)$。

S6 停机。（M为G^*的邻接矩阵）。 □

注意：矩阵M（当算法开始执行S3时，$i=p$）为G^{p-1}的邻接矩阵。此外，处理对角元素不增加新的非零元素。

现在按顺序可得出以下几点：

1. Warshall算法将图G的邻接矩阵M，通过在其上适当改写，则转变成为G的传递闭包的邻接矩阵。正因如此，本算法才被说成是“在原位”工作。
2. 本算法先要处理完指向一个顶点的所有关联边后，才开始处理下一个顶点。换言之，它是逐列处理矩阵M的。因此，我们把 Warshall 算法描述成列-定向的。
3. 处理一个顶点时，没有把指向该顶点的新关联边（即在顶点处理开始时不存在的边）加到该图上。这就意味着：处理一个顶点时，我们可以用任意顺序选择指向该顶点的关联边。
4. 假设指向顶点i的关联边（j, i）在处理顶点i时是不存在的，但是在其后处理某个顶点k时才加进去，$k>i$。很显然，在处理顶点i时，这条边未被处理，此后，因为没有一个顶点作超过一次的处理，所以这条边也不再处理。事实上，这样一条边也不会增加任何新的边。
5. Warshall算法之所以称为以一次通过方式工作，是因为每个顶点恰好处理一次。

假设我们希望修改 Warshall 算法并使其成为行-定向的。在行-定向的算法中，处理一个顶点时，从该顶点引出的所有关联边都要处理。边（i, j）的处理，对于每一条从顶点j引出的关联边（j, k），都要引入边（i, k）。因此，按行处理一个顶点时，就会增加从一顶点引出的新的关联边。其中某些新加的边，在当所考虑的顶点尚未处理完之前，也不予以处理。如果为了计算传递闭包必须对这些边进行处理，那么这种处理只能以二次通过的方式来进行。因此，一般说来，要计算传递闭包，一个行-定向的算法可能需要一次以上的通过。

例如，考虑图14.3（a）的图G。在按行处理G的顶点后，我们得到如图14.3(b)所示的图G'。很显然，G'不是G的传递闭包，这是因为边（1，2）尚未加入图中。也许还会注意到：边（1，3）在这次通过中未被处理，这是因为它仅在处理边（1，4）时才加入的。对于边（4，2），情况也相同。

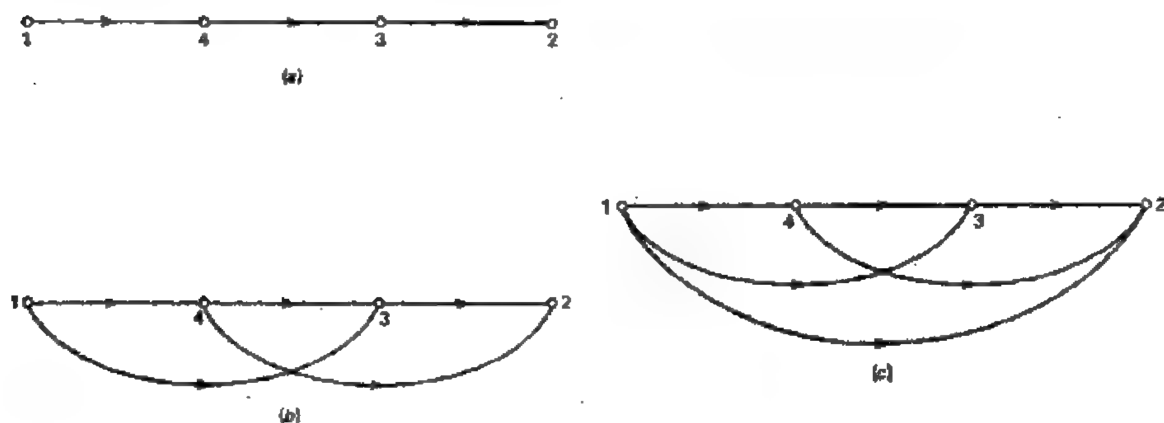


图 14.3 有向传递闭包举例
(a)G; (b)G'; (c)G^*。

假设我们接着来处理G的顶点。在这第二次通过中，边（1，2）在处理顶点1时被加进去了，并得到了图14.3（c）所示的传递闭包G^*。因此，对图14.3（a）中的图G来说，行-定向的算法需要二次通过。

现在产生了一个问题：二次通过是否总能满足。回答是肯定的，并且Warren在文献[14.3]中设计了一个精巧的二次通过的行-定向的算法，并对这一问题进行了证明。在这一算法中，处理一个顶点时，若以顶点i来说，第一次通过时，只处理那些与小于i的顶点相连接的边；第二次通过时，只处理与大于i的顶点相连的边。换言之，该算法是在第一次通过时只对M的主对角线以下的元进行处理，在第二次通过时只对M的主对角线以上的元进行处理后，将图G的邻接矩阵M变换成G^*的邻接矩阵。因此，在每一次通过期间，至多处理$n(n-1)/2$条边。下面给出Warren对Warshall算法修改的描述。

算法14.2 传递闭包（Warren）

$S1$ M为G的邻接矩阵

$S2$ 对于$i=2, 3, \cdots, n$，执行$S3$

$S3$ 对于$j=1, 2, \cdots, i-1$，执行$S4$。

$S4$ 若$M(i, j)=1$，对于$k=1, 2, \cdots, n$，则执行$S5$。

$S5$ $M(i, k)=M(i, k)\vee M(j, k)$。

$S6$ 对于$i=1, 2, \cdots, n-1$，执行$S7$。

$S7$ 对于$j=i+1, i+2, \cdots, n$，执行$S8$。

$S8$ 若$M(i, j)=1$，对于$k=1, 2, \cdots, n$，则执行$S9$

$S9$ $M(i, k)=M(i, k)\vee M(j, k)$

$S10$ 停机（M为G^*的邻接矩阵） □

注意：在上述算法中，$S2$到$S5$步是与第一次通过相对应的，$S6$到$S9$步是与第二次通过相对应的。

举例，重新考虑图14.3(a)所示的图。在Warren算法第一次通过结束时，我们得到了14.4(a)所示的图；在第二次通过结束时，我们就得到了示于图14.4(b)的传递闭包G^*。

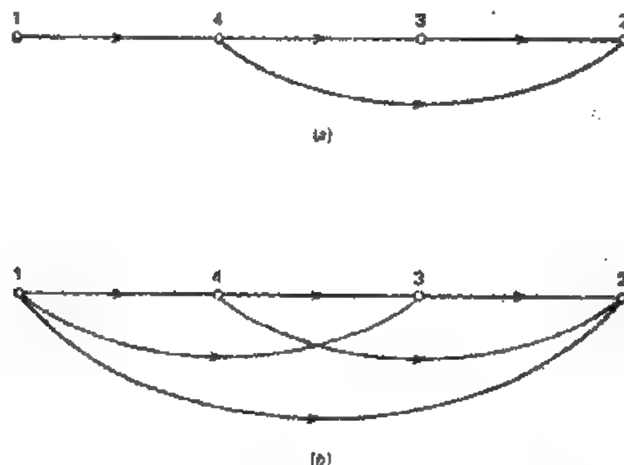


图 14.4 Warren算法的说明

Warren算法正确性的证明是以下面引理为基础的。

引理14.1 假设：对于任意二个顶点s和t，图中存在着一条从s到t的有向路径P，那么在Warren算法的第一次通过（$S2$到$S5$）中，对顶点s处理后所产生的图包含了一条边（s, r），这里r是P中s的一个后继，并且$r>s$或$r=t$。

证明

对s采用归纳法来证明。

如果$s=1$，那么引理显然成立，这是因为P中1的所有后继都大于1。假定引理对所有$s<k$都成立，并令$s=k$。假设（s, i_1）是P中第一条边。如果$i_1>s$，那么引理显然成立。

如果$i_1<s$，那么由归纳法假设可知，在第一次通过时，对顶点i_1处理后所产生的图包含了边（i_1, i_2），这里i_2是P中i_1的一个后继，且$i_2>i_1$或$i_2=t$。

如果$i_2\neq t$且$i_2<s$，那么再由归纳法假设可知，第一次通过时，对顶点得i_2处理后所产生的图包含了边（i_2, i_3），这里i_3是P中i_2的一个后继，且$i_3>i_2$或$i_3=t$。

如果$i_3\neq t$且$i_3<s$，则对i_3重复上面的论证，直到定出一个i_m，并使$i_m>s$或$i_m=t$为止。因此，在处理顶点s之前所具有的图就开始包含了边（s, i_1）、（i_1, i_2）、…、（i_{m-1}, i_m），并满足下列条件：

1.i_p是P中i_{p-1}的一个后继，$p\geq 2$。

2.$i_{m-1}>i_{m-2}>i_{m-3}>\cdots>i_1$，并且对于$k\neq m$，$i_k<s$。

3.$i_m=t$或$i_m>s$。

现在我们开始处理顶点s，由于（i_1, i_2）的存在，（s, i_1）的处理引进了边（s, i_2）。因为，$i_2>i_1$，接着就要处理边（s, i_2）。由于（i_2, i_3）的存在，对该顶点的处理引进了（s, i_3），如此继续下去。最后当完成了对s的处理时，在导出图中就出现了所需要的边（s, i_m）。 □

定理14.2 Warren算法计算了图G的传递闭包。

证明

我们需要考虑两种情形：

情况1　对任意两顶点s和t，G中都存在一条从s到t的有向路径P。

令(i, j)是P中的第一条边(从s到t行进)，并使得$i>j$。那么从上面的引理可推出：在Warren算法第二次通过开始之前，我们所具有的图包含了一条边（i, k），这里k是P中i的后继，且$k=t$或$k>i$。因此，在第一次通过完成之后，则存在一条路径P'：s, i_1, i_2, …, i_m, t，并使得$s<i_1<i_2<\cdots i_m$且每一i_{j+1}都是P中i_j的后继。

在第二次通过中，我们在处理顶点s时，边（s, i_1）是第一次遇到，对边（s, i_1）的处理引进了边（s, i_2），这是由于边（i_1, i_2）存在的缘故。因为$i_2>i_1$，接着就要处理边（s, i_2）。这样，依次引进边（s, i_3），等等。因此，当完成了对s的处理之后，我们就在导出图中得到了边（s, t）。

情形2　G中存在一条包含顶点s的有向回路。

对这种情形，我们可以按上述相同的步骤证明，即在第二次通过中，当完成了对顶点

s 的处理之后，导出图就包含了自环（s，s）。

很显然，Warshall 和 Warren 算法都具有最坏情形下的复杂性 $O(n^3)$。然而，Warren 算法在大的稀疏矩阵下，特别是在耦合环境下的运行速度比 Warshall 算法要快。文献[14.3]还提到了其它行-定向算法。

文献[14.4]已经提出了一种 $O(n^3/\log n)$ 的算法。这种算法是以布尔矩阵相乘的“四名俄国人”算法为根据的。

Strassen 在文献[14.5]中给出了一个对二个 $n\times n$ 矩阵相乘的 $O(n^{2.81})$ 算法。利用这一算法，文献[14.6]和[14.7]对传递闭包问题提出了 $O(n^{2.81}\log n)$ 算法。其它的基于矩阵相乘的算法，可参见文献[14.8]与[14.9]。

文献[14.10]提出了一种算法，这种算法是以 Tarjan 寻求有向图强连通片的算法为根据的（见14.4节）。Munro 算法也是第一步就决定了给定图的强连通片。

尽管 Purdom 算法，见文献[14.11]，在最坏情形下是 $O(n^3)$，但大多数情况是 $O(n^2)$。然而，这一算法比许多其它传递闭包算法要复杂得多。

最近，文献[14.12]提出了一种期望时间为 $O(n+m^*)$ 的算法，这里 m^* 是传递闭包的期望边数。

文献[14.13]讨论了对几个传递闭包算法的体会。

14.2 传递定向

如果我们能对图 G 中的边标定方向，并使得产生的有向图是传递的，那么无向图 G 就是传递可定向的。如果 G 是传递可定向的，那么 $\vec{G}$ 就表示 G 的一个传递方向。

例如，图14.5(a)所示的图就是传递可定向的，该图的一个传递方向，如图14.5(b)所示。

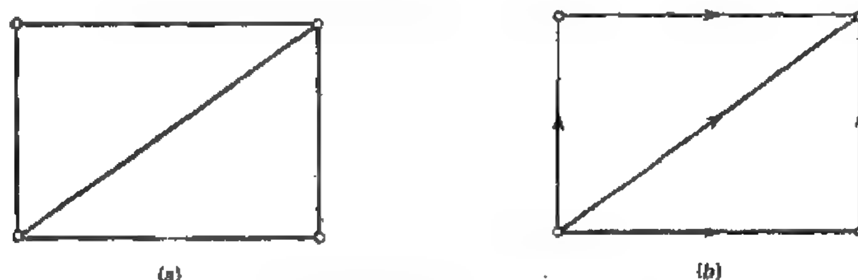


图 14.5 (a)图 G；(b) G 的一个传递方向。

本节，我们讨论一个算法，这一算法是由文献[14.14]给出的，并用来检查一个简单无向图 G 是否为传递可定向的，如果 G 是传递可定向的，则求出一个传递方向 $\vec{G}$。为了算法的展开和描述方便起见，我们引进几个符号：

$i\rightarrow j$ 表示顶点 i 通过从 i 到 j 的定向边与 j 相连。

$i\leftarrow j$ 表示顶点 j 通过从 j 到 i 的定向边与 i 相连。

$i — j$ 表示有一条连接顶点 i 与 j 的边。

$i \not— j$ 表示不存在一条连接顶点 i 与 j 的边。

$i \nrightarrow j$ 表示或者 $i \nrightarrow j$、或者 $i \leftarrow j$ 或者 $i - j$ 未被定向。

$i \rightarrow j$、$i \leftarrow j$、$i - j$ 等也用来表示相应的边。

现在考虑一个传递可定向的无向图 $G=(V, E)$。令 $\vec{G}=(V, \vec{E})$ 表示 G 的一个传递方向。

假设：存在三个顶点 i、j、$k \in V$ 并使得 $i \rightarrow j$、$j - k$ 且 $i \not\sim k$；那么 $\vec{G}$ 的传递性要求 $j \leftarrow k$。类似地，如果对于 i、j、$k \in V$，则有 $i \rightarrow j$、$i - k$ 且 $j \not\sim k$，那么 $\vec{G}$ 的传递性要求 $i \rightarrow k$。

从这两个观察导出了如下简单规则，而这些规则构成了 Pnueli、Lempel 与 Even 算法的基础。

规则R_1 对于 i、j、$k \in V$，如果 $i \rightarrow j$、$j - k$ 且 $j \not\sim k$，那么边 $j - k$ 的定向为 $j \leftarrow k$。

规则R_2 对于 i、j、$k \in V$，如果 $i \rightarrow j$、$i - k$ 且 $j \not\sim k$，那么边 $i - k$ 的定向为 $i \rightarrow k$。

这两条规则示于图14.6，这里虚线表示不存在相对应的边。

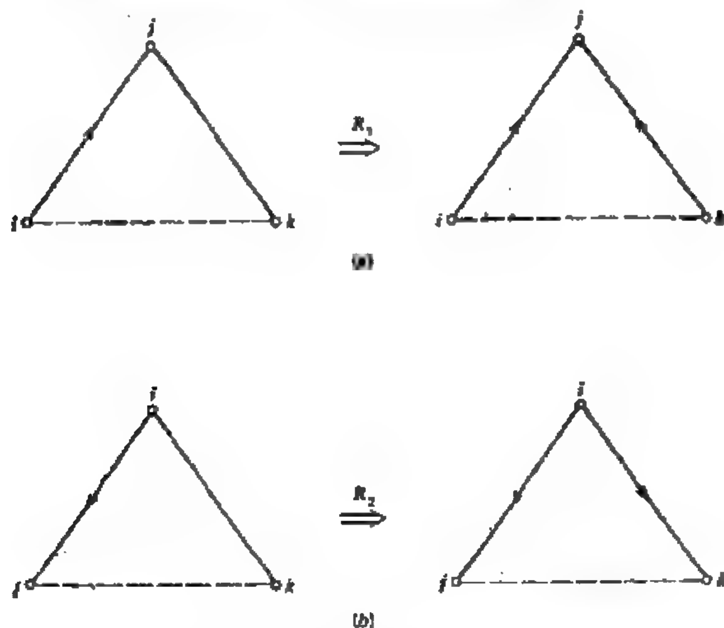


图 14.6 (a)规则R_1；(b)规则R_2。

下面给出一个传递定向算法的描述。

算法14.3 传递定向（Pnueli，Lempel与Even）

S1 G 为给定的简单无向图。置 $i=1$。

S2 （从第i阶段开始）选图G的一条边e，并标记其方向，只要标记是可能，那么利用规则R_1或规则R_2对G中与e邻接的边分配方向。现在，标记有向边e为"已检查。"

S3 检查图G中是否存在一条尚未标记"已检查"的有向边。如果存在，则转到S_4。否则，转到S_6。

S4 令$i \to j$为图中尚未标记"已检查"的有向边。现在对于G中与i或j关联的每一条边来说，只要是可标记的，就照下面的进行，然后置$i \to j$为"已检查"。

情形1 令正在考虑的边为$j - k$。

a.（规则R_1的应用）如果$i \neq k$且边$j - k$未被定向，那么定向$j - k$为$j \leftarrow k$。

b.（与规则R_1的矛盾）如果$i \neq k$且边$j - k$已被定向为$j \to k$，那么产生了一个对规则R_1的矛盾。转到S_9。

情形2 令正在考虑的边为$i - k$。

a.（规则R_2的应用）如果$j \neq k$且边$i - k$未被定向，那么定向$i - k$为$i \to k$。

b.（与规则R_2的矛盾）如果$j \neq k$且边$i - k$已被定向为$i \leftarrow k$，那么产生了一个对规则R_2的矛盾。转到S_9。

S5．转到S_3。

S6．（第i阶段已成功地结束）检查G的所有边是否都已定向。如果已定向，则转到S_8。否则，从G中删去它的所有有向边，并令G'为导出图。

S7．置$G = G'$且$i = i+1$。转向S_2。

S8．（给定图的所有边都已定向，且与规则R_1和R_2相容。这些方向定义为给定图的一个传递方向）停机。

S9．（图G不是传递可定向的）停机。 □

上述算法的主要一步是S_4。在这一步中，我们检查了与一有向边（比如说$i \to j$）邻接的每一条边。如果这样的边已被定向，则检查它的方向以及$i \to j$的方向是否与规则R_1或规则R_2相容。如果正在检查的边尚未定向，那么有可能的话，就利用规则R_1和规则R_2给它分配一个方向。

正如我们所见，算法包括了不同的阶段。每一阶段包括执行S_2，以及随着越来越多的边被定向，还要重复执行S_4。如果一个阶段在没有检查出与规则R_1或R_2的任何矛盾就结束了，那么就意味着：应用这二个规则，在这一阶段中没有更多的边可被定向，并且在这一阶段中已经定向的所有边都与这些规则相容。

算法终止于下面二者之一：（1）检验出规则R_1或规则R_2的一个矛盾；（2）对给定图的所有边分配方向，并使得这些方向与规则R_1和R_2相容。在前一种情况下，图不是传递可定向的；而在后一种情况下，图是传递可定向的，从中导出的有向图确定了一个传递方向。

算法的复杂性取决于执行S_4的复杂性。这一步至多执行m次，这里m是给定图的边数。每次执行S_4都包括检验所有与一有向边相邻接的边。因此，执行S_4所需要的操作量与2Δ成比例，这里Δ是给定图的最大度数。从而算法的总复杂性是$O(2m\Delta)$。

下面举两个例子来说明本算法。首先考虑图14.7(a)所示的图G。

第一阶段 开始将边$7 - 2$定向为$7 \to 2$。按规则R_2，$7 \to 2$蕴含着$7 \to 4$与$7 \to 5$，而$7 \to 4$蕴含着$7 \to 6$。规则R_1是不能应用到与$7 \to 2$邻接的任何边。

按规则R_2，边$7 \to 5$与$7 \to 6$分别蕴含着$7 \to 1$与$7 \to 3$。在这一阶段中，再没有

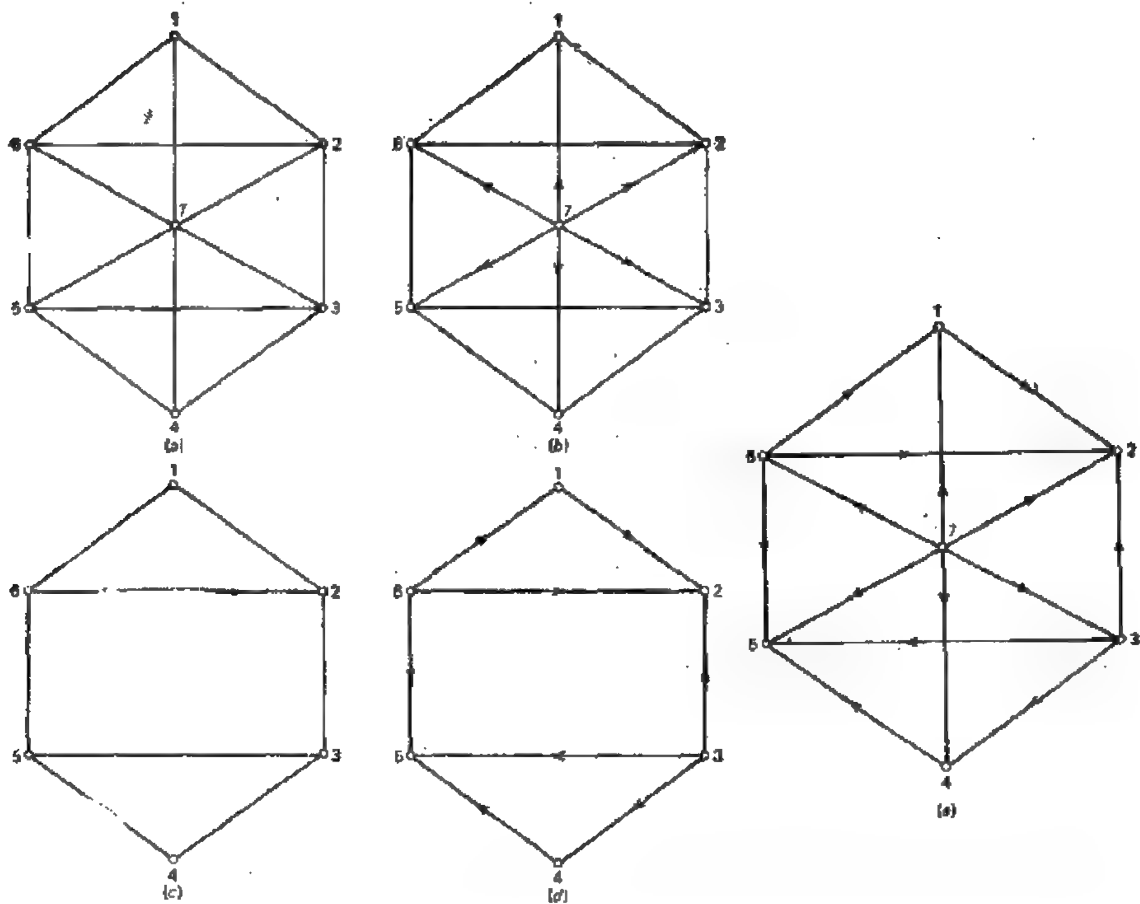


图 14.7

边可被定向了。我们也能检验出：所有分配的方向都是与规则R_1和R_2相容的。现在第一阶段已完成，在这一阶段中被定向的边，如图14.7(b)所示。

现在从图14.7(a)的图G中删去所有在第一阶段中定了向的边，由此得出的图G'，如图14.7(c)所示。第二阶段现在开始，G'为正在考虑的图。

第二阶段　开始将边1—2定向为1→2，这就产生了下面的蕴含序列：

$$(1\to2)\underset{R_1}{\Longrightarrow}(2\leftarrow3)\underset{R_2}{\Longrightarrow}(3\to5)\underset{R_1}{\Longrightarrow}(6\to5)\underset{R_1}{\Longrightarrow}(4\to5)$$

$$(2\leftarrow3)\underset{R_2}{\Longrightarrow}(3\to4)$$

$$(6\to5)\underset{R_2}{\Longrightarrow}(6\to1)$$

$$(6\to5)\underset{R_2}{\Longrightarrow}(6\to2)$$

因此，G'的所有边现在都已被定向，如图14.7(d)所示，G'的这些方向与规则R_1和R_2也是相容的。从而第二阶段成功地结束了。

由此得出图G的传递方向示于图14.7(e)。

其次考虑图14.8所示的图。开始将边1—2定向为1→2，这就产生了下列蕴含序列：

$$(1\to2)\underset{R1}{\Longrightarrow}(2\leftarrow3)\underset{R2}{\Longrightarrow}(3\to4)\underset{R1}{\Longrightarrow}(4\leftarrow5)\underset{R2}{\Longrightarrow}(5\to1)$$

$$\underset{R1}{\Longrightarrow}(1\leftarrow2)$$

以上要求 1 — 2 被定向为 1 ← 2，这与我们分配给边 1 — 2 的方向相矛盾。这样就观测到了与规则$R1$的一个矛盾。因此，图14.8的图不是传递可定向的。

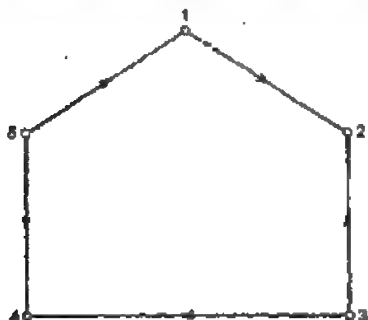


图 14.8

现在我们着手来证明算法 14.3 的正确性。为此，需要证明下面两个断定：

断定 1 如果算法14.3成功地结束(步骤$S8$)，那么由此得出的有向图就是给定图的一个传递方向。

断定 2 如果给定图是传递可定向的，那么算法14.3就会成功地结束。

我们先考虑断定 1。

给定一无向图 $G=(V,E)$。假定算法成功地结束。

现在考虑在算法的相同阶段中被定向的任意二条边 $i\to j$ 和 $k\to l$。那么我们可以构造一个蕴含序列，它起始于有向边 $i\to j$，终止于把边 $k — l$ 定向为 $k\to l$ 的边。这样的一个序列将被称为从 $i\to j$ 到 $k\to l$ 的导出链。注意，从 $i\to j$ 到 $k\to l$ 的导出链可能不只一个。例如，在图14.7(d)的有向图中，下面是从 2 ← 6 到 3 → 4 的两个导出链：

$$(2\leftarrow6)\Longrightarrow(3\to2)\Longrightarrow(3\to4)$$

$$(2\leftarrow6)\Longrightarrow(5\leftarrow6)\Longrightarrow(3\to5)\Longrightarrow(3\to2)\Longrightarrow(3\to4)$$

因此，很清楚，在算法14.3的相同阶段里，讨论一对定向有向边之间的最短导出链是很有意义的。断定 1 的证明是以下面的重要引理为依据的。

引理14.2 在成功地执行第一阶段后，不可能再有三个顶点 i、j、k，从而使 $i\to j$ 且 $j\to k$ 以及 $i\not\to k$。

证明

注意："$i\not\to k$" 表示或者 $i\not\to k$，或者，$i\leftarrow k$，或者边 $i — k$ 未被定向。

显然存在一条连接 i 和 k 的边，否则，我们就会得出一个矛盾，这是因为按规则 $R1$，$i\to j$ 蕴含了 $j\leftarrow k$。

现在假设在算法14.3的第一阶段中，存在形如 $i\to j$，$j\to k$ 而 $i\not\to k$ 的禁止状态。那么从所有引起禁止状态的导出链中选择一条最短链，这条最短链具有最少的、指向 k 的有向边。很显然，任一这种链的长度至少为 3 。令这样的一条链如下：

$$(i\to j)\Longrightarrow(\alpha_1)\Longrightarrow(\alpha_2)\Longrightarrow\cdots\Longrightarrow(\alpha_{p-1})\Longrightarrow(j\to k)$$

现在α_{p-1}对于某些j'，要么是$j\to j'$；对于某些k'要么是$k'\to k$。因此，我们需考虑两种情形。

情形 1 令α_{p-1}为$j\to j'$。

那么蕴含关系$(\alpha_{p-1})\Longrightarrow(j\to k)$就要求$j'\not\to k$，而且$i\to j'$。否则，导出链

$$(i\to j)\Longrightarrow(\alpha_1)\Longrightarrow\cdots\Longrightarrow(\alpha_{p-1})\Longrightarrow(j\to j')$$

就会导致禁止状态$i\to j$、$j\to j'$而$i\not\to j'$的出现。但这条链比我们的导出链更短，这与其最小性相矛盾。

从上述论证中产生出来的这种状态示于图14-9(a)，这里虚线表示相对应的边不存在。

现在按规则R2，$i\to j'$及$j'\not\to k$蕴含了$i\to k$。但是这与我们的假设$i\not\to k$相矛盾

情形 2　令α_{p-1}为$k'\to k$。

与情形 1 相似，蕴含关系(α_{p-1})$\Longrightarrow$($j\to k$)要求$j\neq k'$。而且$i\to k'$。否则，$k'\to k$就会蕴含$i\to k$，这与我们的假设$i\not\to k$相矛盾。此外，$i\to j$与$j\neq k'$蕴含了$i\to k'$。

从上述论证产生的情形，示于图14.9(b)。

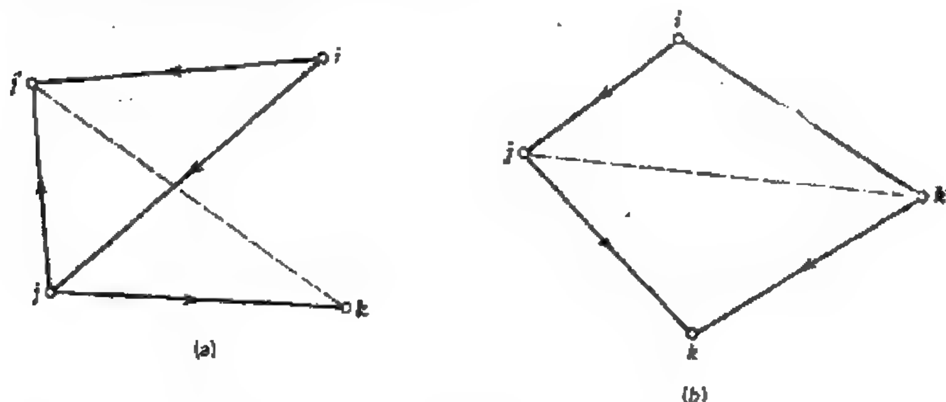


图 14.9

这时，引出了这种禁止情形$i\to k'$、$k'\to k$而$i\not\to k'$的导出链

$$(i\to k')\Longrightarrow(i\to j)\Longrightarrow(\alpha_1)\Longrightarrow\cdots\Longrightarrow(\alpha_{p-1})\Longrightarrow(k'\to k)$$

与我们的导出链有相同的长度，但是进入k的边比我们的要少一。

这又与我们选择链时所作的假设相矛盾。

令E'为在第一阶段中被定向的边集，$\vec{E}'$为相应的有向边集，下面的结论是引理14.2的直接推论。

定理14.3　集合$\vec{E}'$的有向边的子图$\vec{G}_1$(V，$\vec{E}'$)是可传递的。 □

定理14.4　如果算法14.3成功地终止，那么就给出一个可传递定向。

证明

证明是对算法的阶段数进行归纳。如果算法在阶段 1 中将给定图的所有边都确定了方向，那么按照定理14.3，得出的方向就是可传递的。

令给定图$G=(V, E)$在p阶段被定向，令E'为在第一阶段被定向的边集，那么按照定理14.3，子图$\vec{G}'=(V, \vec{E}')$是可传递的。进而，因为子图$G''=(V, E-E')$在$p-1$阶段是可定向的，那么由归纳假设可知，$\vec{G}''=(V, \vec{E}-\vec{E}')$是可传递的。现在我们来证明有向图$\vec{G}=(V, \vec{E})$是可传递的。

假设$\vec{G}$不是传递的，即$\vec{G}$中存在三个顶点i，j，k，从而使$i\to j$与$j\to k$，但$i\not\to k$。那么$i\to j$与$j\to k$不可能同时属于$\vec{E}'$，或同时属于$\vec{E}-\vec{E}'$，这是因为$\vec{G}'$和$\vec{G}''$都是可传递的。

不失一般性，假定$i \to j$在$\overrightarrow{E'}$中，$j \to k$在$\overrightarrow{E}-\overrightarrow{E'}$中，那么必定存在一条连接$i$与$k$的边$i—k$；否则，按规则$R_1$，$i \to j$就会蕴含$j \leftarrow k$。

假定在第一阶段中，边$i—k$被定向为$i \leftarrow k$，那么$\overrightarrow{G}$就不是可传递的，这就与假设产生了一个矛盾。另一方面，如果在以后的阶段中被定向为$i \leftarrow k$，那么$\overrightarrow{G''}$是不可传递的，这又产生了矛盾。

因此，在G中不可能有三个顶点i、j、k，从而使$i \to j$、$j \to k$、但$i \not\to k$，因而，$\overrightarrow{G}$是可传递的。 □

这样就建立了断定1。

下面我们着手建立断定2。

考虑一个传递可定向的图$G=(V, E)$。很显然，如果我们在G的任一传递方向中把所有边的方向都反过来，那么得到的有向图还是G的一个传递方向。

假定我们取G的一条边，并给它标定一个任意的方向，令这条边为$i \to j$。如果我们现在着手应用规则R_1和R_2对其它边标定方向，那么这样定向的边在所有可能的传递方向上具有相同的方向，其中边$i—j$被定向为$i \to j$。这是因为$i—j$的方向一旦被定下来，那么由规则R_1和R_2所导出的方向就必须具有传递可定向性。因此可得出，如果我们对传递可定向的图G应用算法14.3，那么阶段1就会成功地终止，而且不会遇到任何与规则R_1或规则R_2的矛盾。进而在第一阶段被定向的边与G的某些传递方向具有相同的方向。

如果我们能够证明图$G''=(V, E-E')$也是传递可定向的，这里E'是第一阶段中被定向的边集合，那么就可推出：第2阶段以及所有其它阶段都会成功地终止，并给出图G的一个传递方向。从而证明断定2与建立$G''=(V, E-E')$的传递可定向性是相同的。为了这个目标，我们进行如下工作。

设集合E'的边称为标志边，并且把这些边的端顶点称为标志顶点。令V'表示标志顶点集。注意，一条未标志边也可与某一标志顶点相关联。

引理14.3 不可能有三个标志顶点i、j、k，从而使边$i—j$与$j—k$是未标志的，而边$i—k$是标志的。

证明

假设引理中所提到的那类禁止情形存在，即假设存在三个标志顶点i、j、k，从而使边$i—j$和$j—k$未标志，而$i—k$是标志的。则对于这样的三顶点i、j、k，因为j是一个标志顶点，对于某个l则存在一标志边$i—l$。因此，存在一条从标志边$i—k$到标志边$i—l$的导出链。

现在选择一个用i、j、k表示标志顶点，并使得存在一条从$i—k$到$j—l$的导出链P的禁止情形。导出链P是导致禁止情形的所有链中最短的链。这一禁止情形如图14.10(a)所示，边上的楼形表示这条边是标志边，虚线表示不存在相对应的边。

在最短链P中，对某个p来说，紧接$i—k$后面的标志边，要么是$i—p$，要么是$k—p$。不失一般性，我们假定为$k—p$。因此有$i \neq p$，否则，边$k—p$就不能由$i—k$来标志。进而，存在一条连接i和l的边$i—l$，否则边$i—j$就会被标志。因此p和l是分离的。此外还存在边$j—p$，否则边$j—k$就会被标志。到此所建立的关系，如图

14.10(*b*)所示。这时 $j-p$ 就不可能是一条标志边，因为标志它就会导致标志 $i-j$。现在，我们得到一条较短的从边 $k-p$ 到边 $j-l$ 的导出链，从而导致了另一禁止情形，即边 $k-j$ 与 $j-p$ 未被标志，而边 $k-p$ 是标志的。这与假设相矛盾。 □

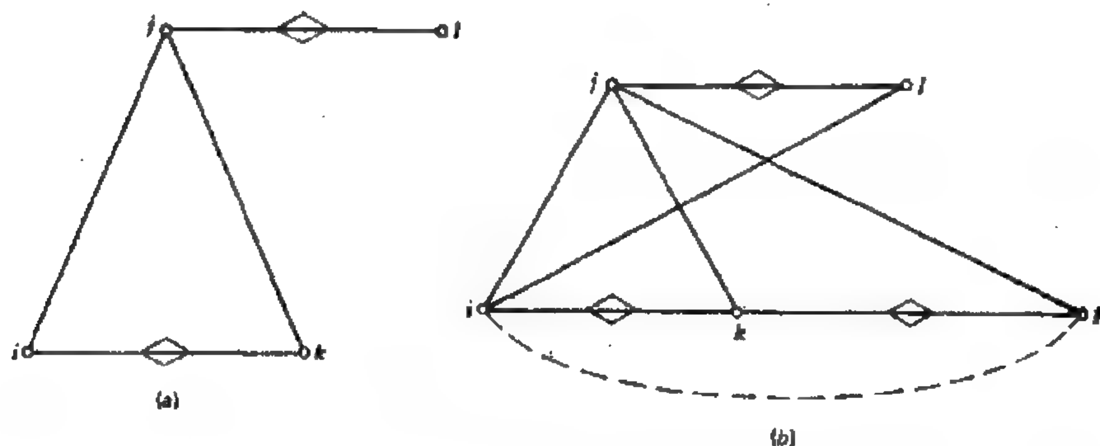


图 14.10

定理14.5 如果$G=(V, E)$是可传递定向的，那么$G''=(V, E-E')$也是可传递定向的。

证明

因为规则$R1$与$R2$只标志了邻接边，所以图$G'=(V', E')$是连通的。

现在考虑一个顶点$v\in V-V'$。如果v是与任一顶点$v'\in V'$相连的，那么v就应与V'中所有与v'相邻接的顶点相连，否则边$v-v'$就会被标志。因为图G'是连通的，所以就得出v必须与V'中所有的顶点相连。

令$\vec{G}$为G的一传递方向，从而使$\vec{E'}$中边的方向与$\vec{G}$中的一致。

现在，我们将集合$V-V'$划分成如下四个子集：

$$A=\{i\,|\,i\in V-V' \text{且在} \vec{G} \text{中对所有} j\in V',\ i\to j\}$$

$$B=\{i\,|\,i\in V-V' \text{且在} \vec{G} \text{中对所有} j\in V',\ j\to i\}$$

$$C=\{i\,|\,i\in V-V' \text{且在} \vec{G} \text{中对所有} j\in V',\ i\nleftrightarrow j\}$$

$$D=V-(V'\cup A\cup B\cup C)$$

注意：D是由$V-V'$中那些与V'中全部顶点相连的所有顶点组成，但并不是D中一个顶点连接到V'中顶点的所有边都确定为相同的方向。

$\vec{G}$的传递性蕴含了V的如下各不同子集间的联系：

1.对于每个$i\in A$，$j\in D$，$k\in B$，有$i\to j$，$j\to k$，$i\to k$。

2.对于所有$i\in C$以及$j\in D$，有$i\nleftrightarrow j$。

3.连接A与C的所有边都被定向为从A到C。

4.连接B与C的所有边都被定向为从C到B。

到目前为止的情形示于图14.1(*a*)。

现将从V'到D定向的所有边的方向反过来，从而使连接V'与D的所有边被定向为从D到V'。这样得出的方向如图14.11(*b*)所示。我们断言：这种方向是可传递的。

为了证明这个断言，我们必须证明：在图14.11(*b*)所示图中，对于所有i、j和k，如果$i\rightarrow j$且$j\rightarrow k$，那么$i\rightarrow k$。很显然，如果这三条边都不在刚被反向的那些边中，这个断言是对的。因此，我们只需考虑下列四种情形：

1.$i\in D$，$j\in V'$并且$k\in V'$。

2.$i\in D$，$j\in V'$并且$k\in B$。

3.$j\in D$，$k\in V'$并且$i\in A$。

4.$j\subset D$，$k\in V'$并且$i\in D$。

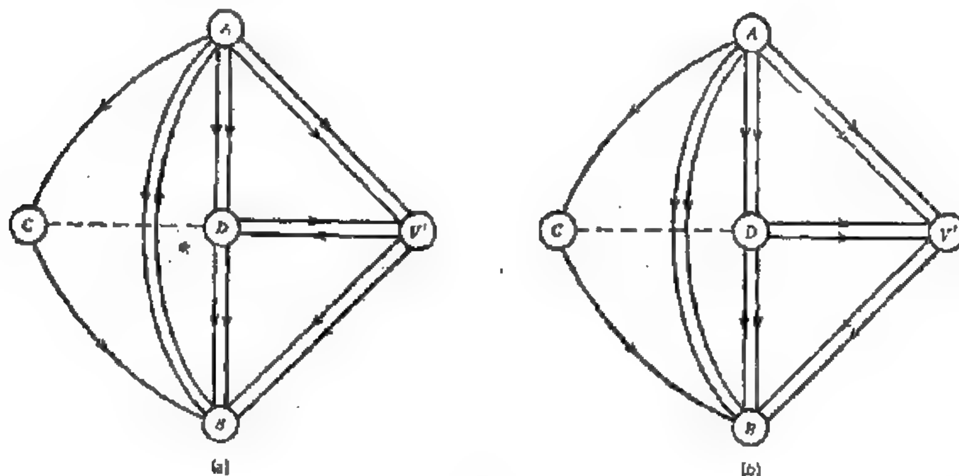


图 14.11

在所有这四种情形下，$i\rightarrow k$示于图14.11(*b*)。因此，图14.11(*b*)的方向是可传递的。

现在从图14.11(*b*)所示的图中移去E'中的所有边，即移去E'中所有的标志边。

假设：在导出图中存在顶点i、j和k，并使得$i\rightarrow j$及$j\rightarrow k$，但$i\nrightarrow k$。这里$i\nrightarrow k$仅意味着$i\leftarrow k$，这是因$i\leftarrow k$就意味着图14.11 *b*的方向不是传递的，如果边$i-k$不在$E-E'$中，那就一定在E'中，否则，图14.11(*b*)中的方向就不是可传递的。因此i和k在V'中。

因为在V'之外不存在顶点，即它既有从V'到它的边，又有从它到V'的边，那么j也在V'中。

这样就有了被标志的i、j、k，未标志的边$i-j$和$j-k$，以及被标志的边$i-k$。根据引理14.3，这是不可能的。

因此，从图14.11(*b*)所示图中移去$\overrightarrow{E'}$的边后所得到的有向图是可传递的。所以，$G''=(V, E-E')$是可传递定向的。 □

这样我们就建立了断定2，并由此证明了算法14.3的正确性。

现在应当清楚地知道，上面所讨论的传递方向算法是一个步骤简单、但其正确性证明却是很绕人的算法实例。关于这个问题较早的算法，可参阅文献[14.15]。

文献[14.14和[14.16]引进了转置图，并且建立了这些图与传递可定向图之间的结构关系。他们还讨论了一个用以测试给定图是否为转置图的算法。

有些图论问题，一般说来是很难解决的，但是当涉及的图是传递可定向的，则问题就变得简单了。求最大完备子集和求最小着色问题都是这类问题的实例，这些问题产生于对存贮再分配以及电路布线问题的研究中，参阅文献[14.16]，也可参阅文献[14.17]与[14.18]。

14.3 深度优先搜索

本节，我们介绍一种搜索图的系统方法，这种方法称为深度优先搜索，简称DFS。它在一些有效算法的设计中被证明是非常有用的，有些算法将留在本章的其它小节来讨论。这里我们介绍的算法是以文献[14.19]为依据的

14.3.1 无向图的DFS

首先我们简述无向图的DFS，一开始，我们就假设正在考虑的图是连通的。如果该图不是连通的，那么DFS将在图的每一片上分别完成。我们还假设图中没有自环。

一个无向图G的DFS按如下步骤进行：

在图G中，任选一顶点v，并以v开始搜索。v被称为DFS根的起始顶点，此时说v已被访问过了。

然后我们选择与v关联的边（v，w），并越过这条边去访问w，同时这条边定向为从v到w。这时边（v，w）就说成是被检验，并称之为树边。顶点v称为w的父亲，记为FATHER（w）。

通常，当我们处于某个顶点x时，就会有两种情况出现：

1.如果与x关联的所有边都已检验过，那么我们就返回到x的父亲，并继续从FATHER（x）搜索。顶点x这时就说成是完全扫描过。

2.如果存在与x关联的某些未被检验的边，那么我们选择这样的边（x，y），并定向为从x到y。这时，边（x，y）就说成是被检验。现在有两种情形都需要考虑：

情形1　如果y原来未被访问过，那么我们就越过边（x，y）去访问y，并从y继续搜索。在这种情况下，（x，y）就是一条树边，且x=FATHER（y）；

情形2　如果y原来已被访问过，那么我们就继续选择一条与x关联的其它未检验过的边。在这种情形下，边（x，y）就称为一回溯边。

在DFS期间，无论顶点x是被首次访问，只要x在搜索期间是被访问的第i个顶点，就给它分配一个特定整数DFN(x)，且令DFN(x)=i。DFN称为x的深度优先数。很显然，深度优先数在DFS期间表示顶点被访问的顺序。

当搜索返回到根，并且所有的边都已被访问时，DFS就终止了。

如上述描述可见，DFS将G的边划分成树边和回溯边，很容易证明，树边形成了G的一个生成树。DFS也给G中的边强加方向，所得到的有向图用$\hat{G}$表示。由DFS所引入的树边及其方向，将形成$\hat{G}$的一个有向生成树，这个有向生成树称为DFS树。

注意：一个图的DFS不是唯一的，因为与一个顶点相关联的边可以任意顺序的选取来

检验。

举例，图14.12中表示了一个无向图的DFS。图中，树边用实线表示，回溯边用虚线表示。依次我们给每一个顶点标出它的深度优先数。图中还列出了一张表，用来表示与每一顶点v相关联的边。这张对于顶点v的表称为v的邻接表，并给出了与顶点v相关联边的检验顺序。

现在我们来介绍DFS算法的一个正式描述。在这一描述中，所考虑的图假定都是不连通的。算法中使用的数组MARK表示，对于每一顶点都有一个元素。描述开始时，将图中每一顶点v置MARK(v)＝0，以此表示所有顶点都未被访问过。只要一个顶点被访问过，就在MARK数组中置相应的元素等于1，数组DFN与FATHER的定义同前，TREE与BACK相应地为存贮所生成的树边与回溯边的二个集合。

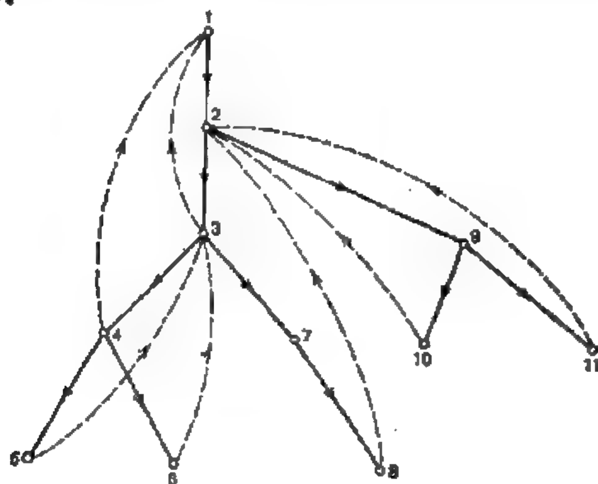
算法14.4　无向图的DFS

S1. G是给定图。置TREE＝∅，BACK＝∅，且i＝1。对G中每一顶点v,置FATHER(v)＝0及MARK(v)＝0

S2.（从G中一个片的DFS开始）选任意一个顶点（比如说顶点r），且MARK(r)＝0。置DFN(r)＝i，MARK(r)＝1，且v＝r。（这个顶点r称为正在考虑片的根）

S3. 如果与v相关联的所有边都已被标记为“已检验”，那么转到S5。（v这时已被完全扫描过）否则选尚未标记“已检验”的边（v，w），从而转到S4。

S4. 边（v,w）定向为从v到w，并作标记“已检验”。照以下步骤进行，然后转到S3。



顶点	相关表
1	(1,2), (1,3), (1,4)
2	(2,1), (2,3), (2,8), (2,9), (2,10), (2,11)
3	(3,1), (3,2), (3,4), (3,5), (3,6), (3,7)
4	(4,1), (4,3), (4,5), (4,6)
5	(5,3), (5,4)
6	(6,3), (6,4)
7	(7,3), (7,8)
8	(8,2), (8,7)
9	(9,2), (9,10), (9,11)
10	(10,2), (10,9)
11	(11,2), (11,9)

图 14.12　一个无向图的DFS

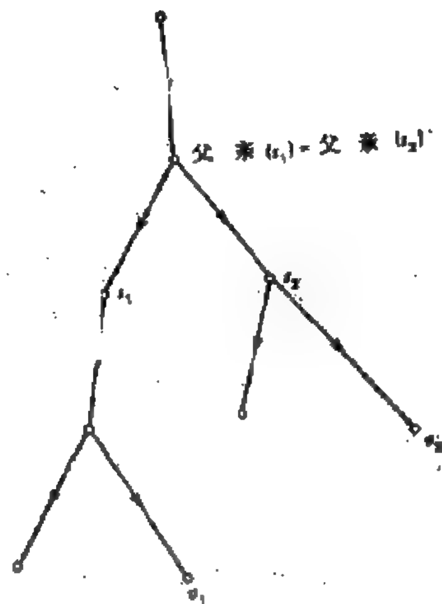


图 14.13

1. 如果MARK(w)= 0，则置

$$i=i+1,$$
$$DFN(w)=i,$$
$$TREE=TREE\cup\{(v,w)\},$$
$$MARK(w)=1,$$
$$FATHER(w)=v,$$
$$v=w$$

2. 如果MARK(w)=1，则置

$$BACK=BACK\cup\{(v,w)\}$$

S5. 如果FATHER(v)≠0（即v不是正在考虑片的根），则置v=FATHER(v)，从而转到S3。否则，转到S6。

S6. 如果对于每一个顶点x，MARK(x)= 1，那么转到S7；否则置 $i=i+1$，转到S2。

S7.（DFS完成）停机。 □

令T为无向连通图的一个DFS树。如前所述，T为$\hat{G}$的一个有向生成树。为了进一步讨论，我们需要引进几个术语。

如果在T中存在一条从顶点v到顶点w的有向路径，那么v称为w的一个前代，w称为v的一个后继。进而，如果$v\neq w$，则v称为w的正常前代，w称为v的正常后继。如果(v, w)是T中一条有向边，那么v就称为w的父亲，w称为v的儿子。注意：一个顶点可以有不止一个儿子。顶点v与它的所有后继顶点形成了T的一个子树，并且顶点w为这个子树的根。

两个顶点v和w是相关的，如果其中一个是另一个的后继，否则，v和w就是不相关的。如果v和w是不相关的，且DFN(v)<DFN(w)，那么v就被说成是w的左侧；否则，v就是w的右侧。G中连接不相关顶点的边称为交叉边。

我们现在来证明：图G中没有交叉边。

令v_1和v_2为T中任意两个不相关顶点，那么显然存在二个相异的顶点s_1和s_2，从而使（1）FATHER(s_1)=FATHER(s_2)；（2）v_1和v_2分别为s_1和s_2的后继顶点（见图14.13）。

令T_1和T_2分别表示为T的以s_1和s_2为根的二个子树。为不失一般性，假定DFN(s_1)<DFN(s_2)。那么按DFS算法，T_2中的顶点显然只有在顶点s_1被完全扫描后才能被访问。进而，s_1的扫描只有在T_1中所有顶点都被完全扫描后才能完成。所以不可能存在连接v_1和v_2的边。如若这样的边存在，那么它就会在s_1的扫描完成之前已经被访问。

从而我们有下面的定理。

定理14.6 如果(v, w)是无向连通图G中的一条边，那么在G中的任一DFS树中，v或者是w的后继顶点，或者相反。 □

无向图中不存在交叉边是一个很重要的性质，它是下节讨论求解一个图的2-连通片算法的基础。

14.3.2 有向图的DFS

有向图的DFS与无向图的DFS基本类似，其主要区别是：在有向图的情况下，边只能沿着其方向通过。由于这个限制，有向图G中的边就在G的DFS中被划分为四类（而不是无向图中的二类）。顶点v所遇到的一个尚未检验的边(v, w)将分类如下：

情形1　w未被访问过。

在这种情况下，(v,w)是一条树边。

情形2　w已被访问过。

a.如果w在DFS林中（即树边的子图）是v的一个后继顶点，那么(v, w)称为一前向边。

b.如果w在DFS林中是v的一个前代顶点，那么(v, w)称为一后向边。

c.如果v和w在DFS林中是不相关的，且$DFN(w)<DFN(v)$，那么(v, w)就是一交叉边。注意：不存在具有$DFN(w)>DFN(v)$的交叉边(v, w)，其证明按定理14.6的证明步骤进行。

现在列出一些有用的注意点：

1.一条具有$DFN(w)>DFN(v)$的边(v, w)或是树边，或是前向边。在DFS期间，因为一条树边总是导致一个新的顶点，所以区分树边与前向边是很容易的。

2.一条具有$DFN(w)<DFN(v)$的边或是后向边，或是交叉边。在检验与v关联的边时遇到边(v, w)时，如果w不是完全扫描过的，那么这样的边就是一条后向边。

3.DFS林（即树边的子图）可以是不连通的，即使正在考虑的有向图是连通的。在DFS林的每个片中，要被访问的第一个顶点称为相应片的根。

下面给出有向图的DFS算法的描述。在这个算法中，我们使用一个新数组SCAN，它对于图中的每个顶点有一元素。开始我们对任一顶点v置$SCAN(v)=0$，这表示没有一个顶点新完全扫描过。只要一个顶点被完全扫描过了，那么SCAN数组中相应的值就置为1。如前所示，当我们遇到一条具有$DFN(w)<DFN(v)$的边(v,w)时，如果$SCAN(w)=0$，那么就把它分类为后向边；否则，(v,w)就是交叉边。我们还使用二个数组FORWARD和CROSS来分别存贮前向边与交叉边。

算法14.5　有向图的DFS

S1. G是给定的无自环有向图。置$TREE=\varnothing$，$FORWARD=\varnothing$，$BACK=0$，$CROSS=\varnothing$，且$i=1$，对G中每一个顶点v，置$MARK(v)=0$，$FATHER(v)=0$和$SCAN(v)=0$。

S2.（DFS以一新根开始）任选一顶点，如r，且$MARK(r)=0$。置$DFN(r)=i$，$MARK(r)=1$和$v=r$。

S3.如果所有与v关联的边都已被标记为"已检验"，则置$SCAN(v)=1$，然后转到S5。（v这时是被完全扫描的）否则，选取一条尚未标记"已检验"的边(v, w)，从而转到S4。

S4.标记边(v, w)为"已检验"，按照下面步骤进行，然后转到S3。

1.如果MARK(w)=0，则置

$$i=i+1,$$
$$\text{DFN}(w)=i,$$
$$\text{TREE}=\text{TREE}\cup\{(v,\ w)\},$$
$$\text{MARK}(w)=1,$$
$$\text{FATHER}(w)=v,$$
$$v=w。$$

2.否则置

FORWARD=FORWARD$\cup\{(v,w)\}$，若DFN(w)>DFN(v)

BACK = BACK$\cup\{v,w)\}$，若DFN(w)<DFN(v)及

SCAN(w)=0;

CROSS=CROSS$\cup\{v,w)\}$，对其它情况。

S5.如果FATHER(v)≠0（即v不是一个根），那么置v=FATHER(v)，转到$S3$，否则，转到$S6$。

S6.若对于每个顶点x，MARK(x)=1，则转到$S7$；否则，置$i=i+1$，转到$S2$。

S7.（DFS完成）停机。 □

举例，一个有向图的DFS如图14.14(a)所示。其次对于每个顶点，我们示出其深度优先数。树边以实线表示，其它边以虚线表示。DFS林被分别示于图14.14(b)。

如前所示，如果有向图本身是连通的，那么有向图的DFS林就可能是不连通的，这点也可从图14.14(b)中看到。这样就给我们提出了一个问题，即寻求DFS林为连通的充分条件。下面我们将要证明：一个强连通图的DFS林是连通的。实际上，我们将建立一个更一般的结论。

令T表示一有向图G(V,E)的一个DFS林。令满足$|V_i|\geq 2$的$G_i=(V_i,E_i)$是G的一个强连通片。考虑G_i的任意两个顶点v和w，不失一般性，假定DFN(v)<DFS(w)。因为G_i是强连通的，所以G_i中存在一条从v到w的有向路径P。令x是P中具有最小深度优先数的顶点，令T_x是以x为根的T的子树。注意：交叉边和后向边都是只从子树T中得出的边。因为这些边导出了具有低于DFN(x)的深度优先数的顶点，由此得出：一旦路径P到达T_x中的一个顶点，那么P中所有后继顶点都将在T_x中。特别是w也在T_x中，因此w也是x的一个后继。因为DFN(x)≤DFN(v)<DFN(w)，因此由DFS算法也得出v在T_x中。从而，G_i中任二个顶点v和w都具有共同的前代顶点，该顶点也在G_i中。

从上面我们可得出结论：G_i的所有顶点有一个共同的前代顶点r_i，其中r_i也在G_i中。现在可以看出：在T中G_i的顶点的所有公共前代顶点中，顶点r_i具有最高深度优先数。进而可见，如果v是G_i中的顶点，那么从r_i到v的树通路中的所有顶点都将在G_i中。这样，由V_i导出的T的子图都是连通的。从而我们得出下面的定理。

定理14.7 令$G_i=(V_i,E_i)$为有向图$G=(V,E)$的一个强连通片。如果T是G的一个DFS林，那么由V_i导出的T的子图是连通的。 □

下面是上述定理的一个直接推论。

推论14.7.1 一个强连通图的DFS林是连通的。 □

容易证明，DFS算法14.4和14.5的复杂性都是$O(n+m)$，这里n为图中顶点数，m为边数。

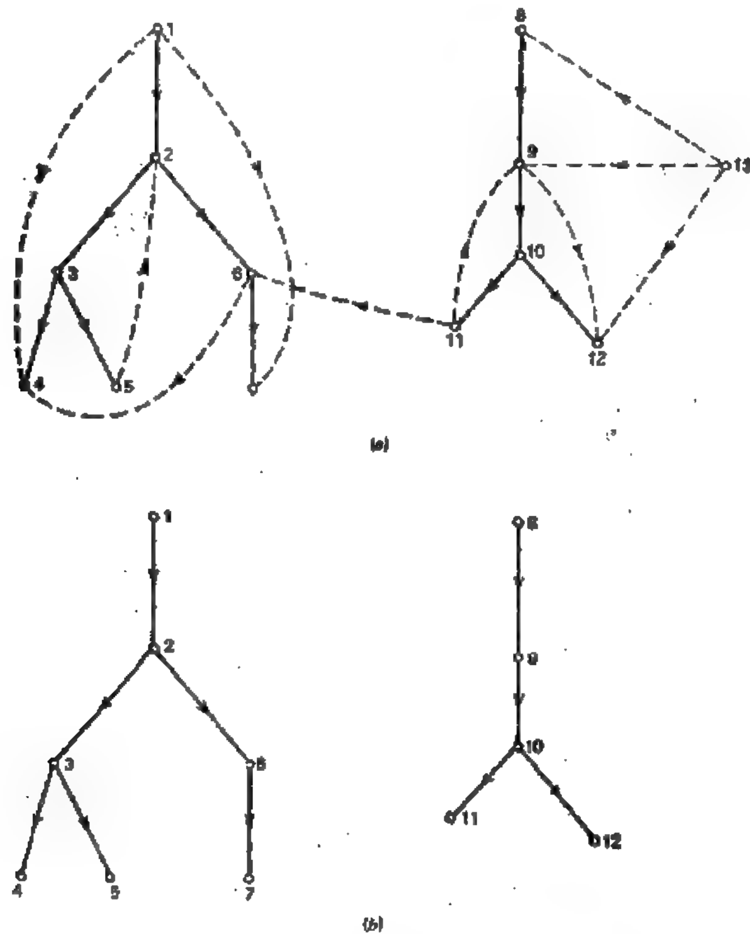


图 14.14 (a)一个有向图的DFS；(b)(a)图中的DFS林。

14.4 2-连通性与强连通性

本节，我们将讨论寻求一个图的2-连通片和强连通片的算法，这些算法是Hopcroft与Tarian在文献[14.20]中和Tarjan在文献[14.19]中提出的。这些算法都是以DFS为基础的。我们先讨论2-连通性算法。

14.4.1 2-连通性

我们曾记得（第八章）：一个2-连通图是没有割点的连通图。图的一个最大2-连通子图称为该图的2-连通片*。

确定2-连通性算法的关键一步就是确定一个简单的判别准则，即在进行DFS时用来认别割点。在下面的两个引理中给出了这样的准则。

* 注意：一个2-连通片与第1.7节定义的块相同。——作者注

令$G=(V,E)$为一无向连通图。令T为G的一个以顶点r为根的DFS树。那么我们有：

引理14.4 顶点$v\neq r$是G的一个割点，当且仅当对v的某个儿子s，T中S的任一后继顶点（包括自身）与v的一正常前代顶点之间不存在后向边。

证明

令G'为从G中移去顶点v后得到的图。按定义，v是G的一个割点，当且仅当G'不是连通的。

令s_1、s_2…、s_k为T中v的儿子。对每个i，$1\leqslant i\leqslant k$，令V_i表示s_i的后继集合（包括本身）并令G_i为在v_i上导出的G'的子图。进而，令$V''=V'-\bigcup_{i=1}^{k}V_i$，这里$V'=V-\{v\}$，并令$G''$为在$V''$上导出的子图。注意：$V$的所有正常前代顶点都在$V''$中。

很显然，G_1、G_2、…、G_k和G''都是G的所有子图，这些子图合起来包含了G'的所有顶点。我们可以很容易地证明：所有这些子图都是连通的。进而，按定理14.6，不存在连接属于不同G_i的顶点的边。所以就得出：G'是连通的，当且仅当对每个i，$1\leqslant i\leqslant k$，在顶点$a\in V_i$与顶点$b\in V''$间存在一条边(a,b)。这样一条边(a,b)必须是一后向边，且b将是v的一正常前代顶点。因此我们得出结论，G'将是连通的，当且仅当对v的每一个儿子s_i，在s_i的某个后继顶点（包括本身）与v的一正常前代顶点之间存在一后向边。引理的证明现在可直接得出。 □

引理14.5 根顶点r是G的一个割点，当且仅当G有不止一个儿子。

证明

证明可按与引理14.4相同的步骤进行。 □

下面我们用深度优先数来表示G的顶点。为了把引理14.4和14.5所给的判别准则嵌入DFS过程中，对于G的每一个顶点v，我们定义：

$\mathrm{LOW}(v)=\min(\{v\}\cup\{w|$ 存在一后向边(x,w)，并使得T中的x是v的后继顶点，而w是v的正常前代顶点。$\})$。

利用上面定义的LOW值，我们可用下面定理来重新描述引理14.4所给出的判别准则。

定理14.8 顶点$v\neq r$是G的一个割点，当且仅当v具有一儿子s，并使得$\mathrm{LOW}(s)\geqslant v$。□

注意，$\mathrm{LOW}(v)$等于最低编号的顶点，该顶点是从v出发，通过一条至少含有一后向边的有向路径而到达的，我们可以重新写式（14.1）为

$\mathrm{LOW}(v)=\min(\{v\}\cup\{\mathrm{LOW}(s)|\ s$是$v$的一个儿子$\}\cup\{w|(v,w)$是一后向边$\})$。

$\mathrm{LOW}(v)$的这一等价定义提供了计算$\mathrm{LOW}(v)$的如下步骤：

1.当v是DFS期间首次访问时，则置$\mathrm{LOW}(v)$等于v的深度优先数。

2.当与v关联的一后向边(v,w)被检验时，则置$\mathrm{LOW}(v)$为当前值与w的深度优先数二者间的最小值。

3.当DFS在v的一个儿子s完全扫描后返回到v时，则置$\mathrm{LOW}(v)$为其当前值与$\mathrm{LOW}(s)$二者中的最小值。

注意：对于任一顶点v，当v的扫描完成时，$\mathrm{LOW}(v)$的计算终止。

下面我们来考虑识别边是否属于2-连通片这一问题。为此，我们采用数组STACK。开始时，STACK是空的，当边被检验时，它们就被加到STACK的栈顶上。

假定v的一个儿子s被完全扫描后，DFS就返回到顶点v。在这里将完成LOW(s)的计算。假定现在发现LOW(s)≥v，那么，按定理14.8，v是一个割点。进而，如果s是具有这种性质的第一个顶点，那么我们容易发现：与s相关联的边（v，s）及其后继顶点将构成一个2-连通片，这些边正好是从STACK栈顶到包括(v，s)在内的那些边。现在将它们从STACK中删去。根据这一点，算法恰好以图G'上相同的方式进行，其中图G'是通过从G中移去刚识别为2-连通片的那些边而得到的。

例如，一个连通图的DFS树可以如图14.15所示。G_1，G_2，…，G_5为按照他们定义的顺序排出的2-连通片。

现在给出2-连通算法的描述。这个算法基本上与算法14.4相同，它包含计算LOW(v)值以及识别属于不同2-连通分支的割点和边的相应步骤。注意，在这一算法中，根顶点r为了识别包含r的2-连通分支，被当作一个割点来处理，尽管r本身不是一个割点。

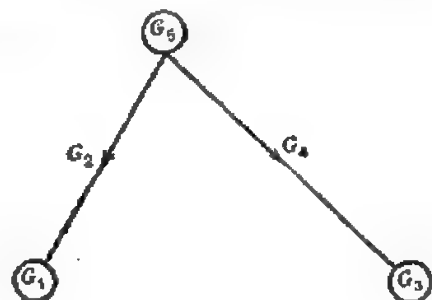


图 14.15 G_1，G_2，G_3，G_4，G_5——一个图的2-连通分支

算法 14.6 2-连通性

S1. G为给定的连通图。对于G中每一顶点v，置FATHER(v)＝0和MARK(v)＝0。

置$i=1$和STACK＝∅。

S2. 任选顶点r，并满足MARK(r)＝0。置

DFN(r)＝i，

LOW(r)＝i，

MARK(r)＝1，

$v=r$。

S3. 如果v的所有关联边都已标记为“已检验”，那么转到S5。否则，选取一尚未标为“已检验”的边（v，w），对这条边作标记“已检验”，再加到STACK的栈顶，并转到S4。

S4. 做如下事情，然后转到S3：

1. 如果MARK(w)＝0，则置

$i=i+1$，

DFN(w)＝i，

LOW(w)＝i，

FATHER(w)＝v，

MARK(w)＝1，

$v=w$。

2. 如果MARK(w)＝1，则置

$$\mathrm{LOW}(v)=\min\{\mathrm{LOW}(v),\ \mathrm{DFN}(w)\}$$

S5. 如果FATHER(v)≠0，则转到S6；否则，转到S8。

S6. 如果LOW(v)≥DFN(FATHER(v))，那么从STACK栈顶移去所有的边，直到

包含了边（FATHER(v)，v）为止（一个2-连通片就被求出来了。）

S7. 置

$$\text{LOW}(\text{FATHER}(v))=\min\{\text{LOW}(v),\ \text{LOW}(\text{FATHER}(v))\},$$

$$v=\text{FATHER}(v)$$

并转到S3.

S8.（所有2-连通片都被求出来了）停机。 □

上述算法的举例说明，见图14.16。

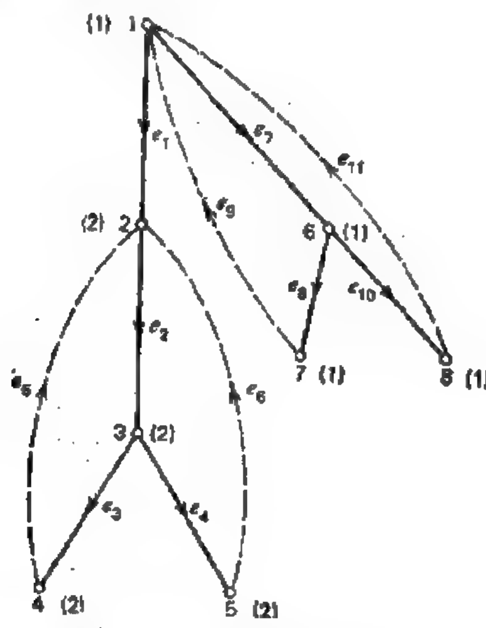


图 14.16 算法14.8的举例说明。LOW值示于括号中。2-连通片为{e_2, e_3, e_4, e_5, e_6}，{e_1}和{e_7, e_8, e_9, e_{10}, e_{11}}。

14.4.2 强连通性

曾记得，如果对于每两个顶点v和w，图G中都存在一条从v到w的有向路径和一条从w到v的有向路径，那末图G就是强连通的。进而，图G的最大强连子图称为该图的强连通片。

考虑一个有向图$G=(V,\ E)$，令$G_1=(V_1,\ E_1)$，$G_2=(V_2,\ E_2)$，…，$G_k=(V_k,\ E_k)$为G的强连通片。令T为G的一个DFS树，而T_1，T_2，…，T_k分别为T在顶点集V_1、V_2、…、V_k上的导出子图。由定理14.7可知：T_1、T_2、…、T_k都是连通的。

令r_i（$1\leqslant i\leqslant k$）为T_i的根。假定：如果$i<j$，那么DFS终止于顶点r_i而先于顶点r_j。那么我们就会看到：对于T中的每个$i<j$，r_i或者是在r_j的左边，或者是r_j的后继顶点。进而对于$1\leqslant i\leqslant k$，G_i可以由那些不在G_1、G_2、…、G_{i-1}中的r_i的后继顶点组成。

推导强连通性算法的第一步，就是确定一个在DFS过程中用来识别根的简单判据。下面的观察在导出这一判据中将是有用的。这些观察都是下面事实的直接推论，即在通过收缩E_1，E_2，…，E_k的每一集合的所有边而得到的图中，不存在有向回路。（见5.1节）

1.不存在（v，w）这类后向边，这里$v\in V_i$，$w\in V_j$，$i\neq j$。换言之，所有的后向边

都是从V_i中的顶点出发，最终仍终止在V_i的顶点上。

2.不存在（v，w）这类交叉边，这里$v \in V_i$，$w \in V_j$，$i \neq j$，r_j是r_i的一个前代顶点。因此，对于每条交叉边（v，w），下面二者之一必为真。

a.对于某个i和j，$i \neq j$且r_j在r_i的左边，则$v \in V_i$且$w \in V_j$。

b.对于某个i，则$v \in V_i$且$w \in V_i$。

假定G的顶点用它们的DFS数来命名，G中的每个v定义为：

LOWLINK$(v)=\min(\{v\} \cup \{w|$ 存在一条从v的后继顶点到w的交叉边或后向边，且w与v在相同的强连通片内$\})$。

假设$v \in V_i$。那么从上述定义可得：LOWLINK(v)是V_i中最小编号顶点，该顶点是从v出发通过至少包含一条后向边或一条交叉边的有向路径而达到的。从上述进行的这些观察可得：这样一条有向路径的所有边都必须在G_i中。作为一个直接的结果我们得出

$$\text{LOWLINK}(r_i)=r_i，\text{对于所有的 } 1 \leqslant i \leqslant k \tag{14.2}$$

假设$v \in V_i$且$v \neq r_i$。那么G_i中存在一条从v到r_i的有向路径P。这样的一条有向路径P必须包含一条后向边或一条交叉边，这是因为$r_i < v$，并且只有交叉边或后向边才能得出较小编号的顶点。换句话说，P包含了一个顶点$w < v$。所以，对于$v \neq r_i$，我们得到

$$\text{LOWLINK}(v) < v \tag{14.3}$$

结合式（14.2）和（14.3），我们得到下面的定理。该定理描述了有向图的强连通片根的特征。

定理14.9 一个顶点v是有向图的强连通片的根，当且仅当LOWLINK$(v)=v$。 □

当我们进行DFS时，可以应用下列步骤来计算LOWLINK(v)。

1.首次访问v时，置LOWLINK(v)为v的DFS数。

2.如果一后向边（v，w）已被检验，则置LOWLINK(v)为其当前值与w的DFS数二者中的最小值。

3.如果一条交叉边（v，w）被搜索，这里w与v在同一个强连通片中，则置LOWLINK(v)为其当前值与w的DFS数二者中的最小值。

1.当搜索在v的一个儿子被完全扫描后返回到v,则置LOWLINK(v)为其当前值与LOWLINK(S)值中的最小值

为了实现第三步,我们需要一次测试,用来检查w是否与v在同一强连通片中。为此,我们用数组STACK1来确定1的哪个顶点能按DFS中被访问的顺序加到STACK1中。STACK1也用来确定顶点是否属于一个强连通片。

令v为DFS期间的第一个顶点，且有LOWLINK$(v)=v$。那么由定理14.9可知，v是一个根，事实上v就是r_1。这里，从STACK1栈顶上的顶点直到包括v的顶点刚好就是属于G_1的那些顶点，从而G_1容易被识别。现将这些顶点从STACK1中删去。这时算法的执行与在图G'上进行的方法相同，其中图G'是通过从G中删去G_1的顶点而得到的。

至于在LOWLINK计算中第三步的实现,令$v \in v_i$，(v, w)为检验v的关联边时遇到的一条交叉边。假设w不与v在同一个强连通片中。那么w就属于一个根r_j为r_i的左顶点(观察2a，P462）的强连通片G_j。这一连通片的顶点也许已被识别过，因而也就不在STACK1

中了，这样，w与v在同一个强连通片中，当且仅当w在STACK1中。

现在给出强连通性算法的一个描述，这个算法基本与算法14.5相同，其中算法14.5只是包含了计算LOWLINK值与识别相异强连通片的顶点所要的相应步骤。在这个算法中，我们使用一个数组POINT。开始时对于每一顶点v，POINT$(v)=0$，以此表示数组STACK1中没有顶点。当v加到STACK1时，置POINT为1，而且当v从STACK1中除去时，置POINT为0。

算法14.7　强连通性

S1. G为给定有向图。对G中每一顶点v，置MARK$(v)=0$，FATHER$(v)=0$，且POINT$(v)=0$。置$i=1$，STACK1$=\varnothing$

S2. 取任一顶点r，及MARK$(r)=0$。置

$$\text{DFN}(r)=i,$$
$$\text{LOWLINK}(r)=i,$$
$$\text{MARK}(r)=1.$$

把r加入STACK1、置POINT$(r)=1$和$v=r$。

S3. 如果所有与v相关联的边都被标记为“已检验”，那么转到S5。否则，取一尚未标记为“已检验”的边(v, w)，把它标记为“已检验”，并转到S4。

S4. 做如下事情，然后转到S3：

1. 如果MARK$(w)=0$，置

$$i=i+1,$$
$$\text{DFN}(w)=i,$$
$$\text{LOWLINK}(w)=i,$$
$$\text{FATHER}(w)=v,$$
$$\text{MARK}(w)=1.$$

把w加入到STACK1，并置POINT$(w)=1$和$v=w$。

2. 如果MARK$(w)=1$，DFN$(w)<$DFN(v)，且

POINT$(w)=1$，那么置

$$\text{LOWLINK}(v)=\min\{\text{LOWLINK}(v),\ \text{DFN}(w)\}$$

S5　如果LOWLINK$(v)=$DFN(v)，那么删去STACK1栈顶的所有顶点并包含v。(这些顶点构成了一强连通片。)然后，对于从STACK1中除去的所有顶点x，置POINT$(x)=0$

S6　如果FATHER$(v)=0$，则转到S7；否则，置

1. LOWLINK(FATHER(v))=
   min{LOWLINK(FATHER(v)), LOWLINK(v)}。
2. $v=$FATHER(v)，

然后转到S3。

S7　如果对于每一顶点x，MARK$(x)=1$，则转到S8；否则转S2。

S8　(所有强连通片都已被识别出来了。)停机。　□

关于上述算法的举例说明，见图14.17。

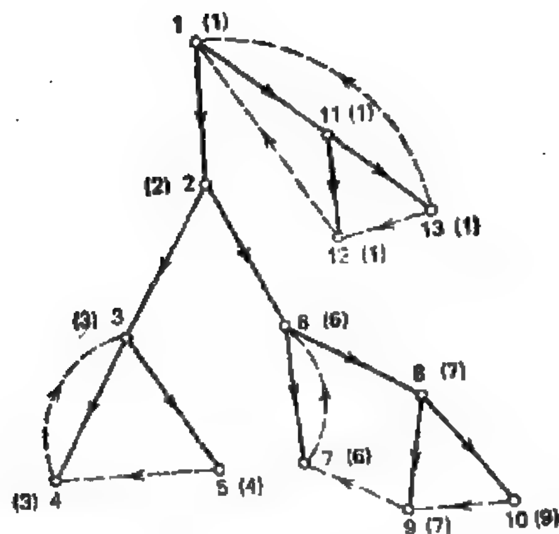


图 14.17　算法14.7的说明。LOWLINK值示于括号中，强连通片为{3, 4, 5}，{6, 7, 8, 9, 10}，{2}和{1, 11, 12, 13}。

14.5　程序图的简化

程序图就是一个具有特殊顶点 s 的有向图 G，其中从 s 到 G 的每个其它顶点都有一条有向路径。换言之，G 的每个顶点都是从 s 出发可到达的。顶点 s 称为 G 的起始顶点。我们假定在程序图中没有平行边。这个假定就本章所涉及的讨论来说不失一般性。

一个计算机程序中的控制流可用程序图来模拟，程序图中的每个顶点代表一个可以顺序执行的指令块。在研究有关编码优化的问题中，计算机程序的这种表示被证明是非常有用的。

为了许多编码优化方法的运用，其程序图就必须有一种叫做可约性的特殊性质。见文献[14.21]到[14.29]。

一个程序图 G 的可约性用下面两个在 G 上的变换来定义：

S_1：删去 G 中的自环 (v, v)。

S_2：如图 (v, w) 是指向 w 的唯一关联边，且 $w \neq s$，则删去顶点 w。对 G 中的每一条边 (w, x)，如果 (v, x) 原先不在 G，那么就增加一条新边 (v, x)。（这个变换被称为把顶点 w 压入顶点 v 中）

例如，在图14.18(a)所示的程序图中，把顶点5压入顶点4中，就导致了图14.18(b)所示的图。

如果一个程序图能够重复应用变换 S_1 和 S_2 来变换成只包含顶点 s 的一个图，这个程序图就是可约的。

例如，图14.18(a)中的图是可约的。可以验证，若按顺序5，8，4，3，10，9，7，6，2来压缩顶点，这个图就可被约简。

Cocke在文献[14.22]和Allen在文献[14.27]中最早提出可约性这一概念的，他们是用所谓区间分析这样一个技术语来定义的。上面所给的定义起因于Hecht和Ullman的文献

[14.30]，并且与Cocke和Allen的定义等价。

如果一个图是可约的，那么可以证明：由G通过一次或多次应用变换S_1和S_2而得到的任何图G'也是可约的。所以，应用变换的次序对检验可约性没有影响。进而，某些如"go-to-less"的有趣程序所产生的图必须是可约简的（见文献[14.30]），并且多数程序利用"节点分裂"的过程都可以用一个可约的图来模拟（见文献[14.31]）。

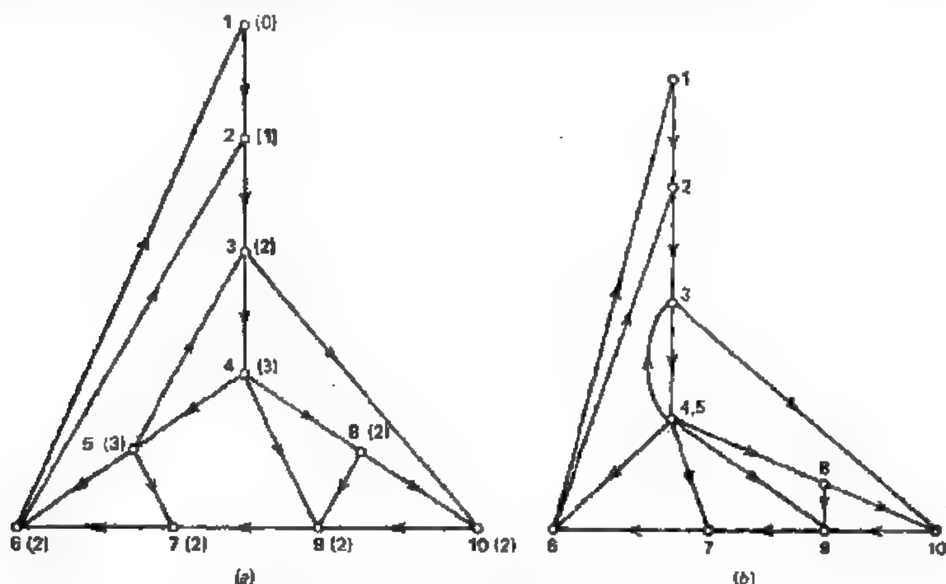


图 14.18 (a)一个可约简的程序图G，HIGHPT1值在括号中给出；(b)在G中把顶点5压入顶点4得到的图。

假设我们希望测试一个图的可约性。这一过程可以首先应用变换S_1删去自环，然后再计数指向每一个顶点的关联边数来完成。其次，我们可以找到一个顶点w，其中只有一条关联边(v, w)指向w，并应用变换S_2，把w压入v中。然后我们重复这一过程，直到该图完全被约简或者发现它是不可再约简时为止。很显然，S_2的每次应用都需$O(n)$时间，这里n为G的顶点数，并且每次将顶点数减1。从而这一算法的复杂性是$O(n^2)$。Hopcroft和Ullman[14.32]中把这个算法改进到$O(m\log m)$，这里m为G的边数。接着Tanjan在文献[14.33]、[14.34]中又给出了一个算法，这个算法比Hopcroft和Ullman的算法更为实用。

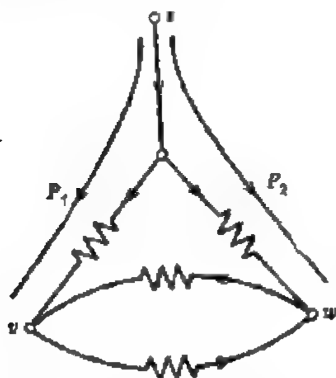


图 14.19 一个基本的不可约简图

Hecht和Ullman在文献[14.30]、[14.35]中给出了程序图的几个有用的结构特征。其中之一由如下定理形式给出。

定理14.10 令G为一个始顶点为s的程序图。G是可约简的，当且仅当不存在相异顶点$v\neq s$和$w\neq s$，不存在从s到v的有向路P_1和从s到w的有向路P_2，以及不存在包含v和w的有向回路C，从而C与P_1和P_2没有共同边但只有一个共同顶点（见图14.19）。□

在文献[14.30]和[14.36]中可以找到上面定理的证明。

本节，我们讨论检测一个程序图的可约性的 Tarjan 算法。这个算法要用到 DFS，并且以可约简程序图的一个特征为根据的，其中的这个特殊将用定理14.10来证明。这里，我们的讨论是以文献[14.33]为基础的。

令 G 是起始顶点为 s 的一个程序图，T 是以 s 为根的 G 的一棵DFS树。今后，我们将用它们的深度优先数来表示顶点。

定理14.11 G 是可约简的，当且仅当 G 不包含从 s 到某个顶点 v 的有向路径 P，从而 v 在 T 中是 P 的某个其它顶点的正常前代顶点。

证明

假设 G 是不可约的，那么存在满足定理14.10条件的顶点 v 和 w，有向通路 P_1、P_2 和回路 C。假定 $v<w$。令 C_1 为 C 中从 v 到 w 的部分。那么 C_1 包含 v 和 w 的某个公共前代顶点 u。因此，这条由 P_2 和 C 的从 w 到 u 的部分得到的有向路径满足定理的条件。

反之，假设存在一条满足定理条件的有向路径 P。然后令 v 为 P 中的第一个顶点，它是 P 中较前的某个顶点的一个正常前代顶点。假设 w 是 P 中 v 的第一个后继顶点。令 P_1 为 T 中从 s 到 v 的部分，P_2 为 P 中从 s 到 w 的部分。同时令 C 为有向回路，这条有向回路包括了 P 中由 w 到 v 部分以及从 v 到 w 的树边组成的路径。这时我们发现，v、w、P_1、P_2 和 C 满足定理14.10的条件。所以，G 是不可约简的。 □

对于任一顶点 v，令HIGHPT1(v)为 v 的最高编号的正常前代，从而存在一条从 v 到HIGHPT1(v)的有向通路 P，并且 P 不包含 v 中除HIGHPT1(v)之外的其它正常前代。如果不存在从 v 到 v 的一个正常前代的有向路径，我们则定义HIGHPT1(v)=0。举例，在图14.18(a)的括号中，我们给出了相应顶点的HIGHPT1值。

注意在HIGHPT1(v)的计算中，我们可能忽略前向边，这是因为，如果 P 是从 v 到 w 的一条有向通路，并且 P 不包含 v 中除了 v 和 w 以外的前代顶点，我们就有可能用树边组成的路径或它的一部分来代替 P 的每条前向边，并且仍有一条从 v 到 w 的有向路径，此路径除 v 和 w 外不包含 v 的前代顶点。

Tarjan算法是以程序图的下列特征为依据的。

定理14.12 G 是可约的，当且仅当不存在具有指向 v 的关联边(u, v)、且 $w<$HIGHPT1(v)的顶点 v，这里 w 是 u 和 v 的最高编号的公共前代顶点。

证明

假定 G 是不可约的，那么根据定理14.11可知，存在一条从 s 到 v 的有向通路，其中 v 为 P 中某个其它顶点的一个正常前代顶点。选择尽可能短的 P。令 w 为 P 中 v 的第一个后继顶点，那么 P 中继 w 之后的、除 v 以外的所有顶点都是 T 中 v 的后继顶点。换言之，P 中从 w 到 v 部分除 v 以外不包含 w 的正常前代顶点。所以 HIGHPT1(w) $\geqslant v$。因此，w，HIGHPT1(w)和 P 中指向 w 的边，都满足定理的条件。

反之，假定定理的条件成立。那么边(u,v)就不是一后向边，因为在这种情况下，u 和 v 的最大编号公共前代顶点 w 是 v，由此 $w=v\geqslant$HIGHPT1(v)。因此(u, v)或者是一条前向边，或者是一条交叉边，令 P_1 为一条从 v 到 HIGHPT1(v)的有向路径，其中这条路径除了HIGHPT1(v)以外不再通过 v 的正常前代顶点。那么包含从 s 到 u 的树边以及由边

(u, v)再接P_1的有向路径满足定理14.11的条件。所以，G是不可约的。 □

现在很清楚，应用上述定理来检测一个图G的可约性，包括下面几个主要步骤：

1.完成G的DFS，s为根顶点。

2.对于G中每个顶点v，计算HIGHPT1(v)。

3.在HIGHPT1的计算过程中，对于交叉边，检验定理14.12的条件。

4.在计算HIGHPT1之后，对于前向边，检验定理14.12的条件。

注意，如前所见，在计算HIGHPT1过程中，可以忽略前向边。进而，正如证明定理14.12时所看到的那样，检验定理14.12的条件不需要检测后向边。

为了计算HIGHPT1值，首先将边$(u、v)$按v的编号排序，然后我们按照从最高到最低v的顺序处理后向边。开始时，所有顶点都未标记。为了处理后向边(u, v)，对于每个当前未标记的顶点v，我们从u到v的树路径向上作标记。(我们不对v本身作标记。)如果一个顶点w作标记了，那么就来检验指向w的所有交叉边。如果(z, w)是这样一条交叉边(见图14.20)，对于每个未标记的顶点，我们就从z到v的树通路方向上作标记v。如果z不是v的一个后继顶点，那么根据定理14.12，G就是不可约的，并且计算停止。我们继续作标记，直到用完指向已标顶点的交叉边为止。然后我们再处理后一个下向边。当所有后向边都已处理过了，标记也就给出了顶点的HIGHPT1值。每个未标顶点的HIGHPT1值等于0。

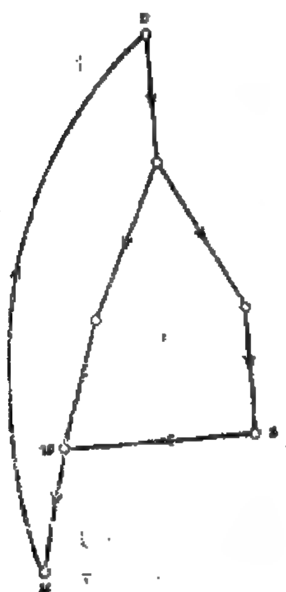


图 14.20

现在我们来叙述检测可约性的Tarjan算法。本算法中，我们使用称为存贮桶的n个队列，每个顶点有一个队列。顶点w相应的存贮桶BUCKET(w)包含一张表，表中元素为指向顶点w的后向边(u, w)。在处理后向边(u, w)时，我们需要记录一些顶点，从这些顶点出发经过有向路径就可到达u。为此，使用集合CHECK。

算法14.8 程序图的可约性(Tarjan)

S1.完成给定n-顶点程序图G的一次DFS。用它们的DFS数表示顶点。按v的编号对G的后向边(u, v)排序。

S2.对于$i=1, 2, \cdots, n$，置

HIGHPT1(i)＝0，

BUCKET(i)＝空表。

S3.把每一后向边(u, w)加到BUCKET(w)中。

S4.置$w=n-1$。

S5.检测BUCKET(w)是否为空的。如果是空的，则转到$S6$；否则转到$S7$。

S6.置$w=w-1$。如果$w<1$，则转到$S16$；否则，转到$S5$。

S7.(从处理一条新的后向边开始)从BUCKET(w)中删去一后向边(x, w)，并

置CHECK = $\{x\}$。

S8. 检测 CHECK 是否为空的。如果是空的（一条后向边处理结束），则转到S_5；否则，转到S_9。

S9. 从CHECK中删去u。

S10. 检测u是否是w的一个后继顶点。如果是一个后继顶点，则转到S_{11}；否则，转到S_{17}。

S11. 检测是否$u=w$。如果是，则转到S_8；否则，转到S_{12}。

S12. 检测是否HIGHPT1$(u)=0$。如果是，则转到S_{13}；否则，转到S_{15}。

S13. 置HIGHPT1$(u)=w$。

S14. 对于每个交叉边(v, u)，把v加到CHECK中。

S15. 置$u=$FATHER(u)，转到S_{11}。

S16. 对于每一条前向边(u, v)，如果$u \geqslant$HIGHPT1(u)（这个图是可约的），那么就停机。否则，转到S_{17}。

S17. 停机。这个图是不可约的。 □

注意，算法14.8的S10要求能够确定一个顶点w是否是另一个顶点u的后继顶点。令ND(u)为T中顶点u的后继顶点数。那么我们就可以证明：w是u的一个后继顶点，当且仅当$u \leqslant w < u+$ND(u)（见习题14.3）。在DFS期间，我们可以用简单的方法计算ND(u)。

算法14.8的有效性关键取决于HIGHPT1计算的有效性。为了使HIGHPT1计算是有效的，在算法14.8中，我们要避免检验已标记过的顶点。Tarjan提供了实现这一点的一个方法。下面的观察形成了这一过程的基础。

假设进行到S12，我们正在检验一顶点u的标记。在这个阶段，令u'为u的最高未标记的正常前代顶点。这就意味着除了u'以外的u的所有正常前代顶点都已被标记，这些顶点在T中处于u'与u之间。从而u'是下一个要检验的顶点。进而，当u已被标记时，所有顶点（本来以u为其最高未标记的正常前代顶点）将把u'作为最高未标记的正常前代顶点。

为了实现从上面观察得出的方法，我们将使用编号从1到n的集合。如果v是w的编号最高未标记的正常前代顶点，那么一个顶点$w \neq 1$就放在编号为v的集合中，即SET(v)。因为顶点1永远不会被标记，所以每个顶点总是在一个集合中。开始时，一个顶点是在T中编号为它父亲的集合中。因此，开始时，对于$i=2, 3, \cdots, n$，把i加到SET(FATHER(i))中。

为了进行S_{15}，我们找出了编号为u'的包含u的集合，并令其为新的u。进而，当u被标记时，我们将编号为u和u'的集合合并以形成一个新编号为u'的集合。从而u'就成为原来集合SET(u)中所有顶点的未标记正常前代顶点的最高编号。

现在可见：用下面一系列步骤来代替算法14.8的S_{12}到S_{15}，就会实现上面描述的方法。当然，如前所述，SETs要进行初始化。

S12'. 置$u'=$包含u的集合编号。

S13'. 检测是否HIGHPT1$(u)=0$。如果是，则转到S_{14}'；否则，转到S_{16}'。

S14'. 置

1. $\text{HIGHPT1}(u)=w$; 并且

2. $\text{SET}(u')=\text{SET}(u)\cup\text{SET}(u')$。

S15'. 对于每一条交叉边 (v, u)，把v加到CHECK中。

S16'. 置$u=u'$。转到$S11$。

可以证明：如上所修改的HIGHPT1计算，需要$O(n)$次集合的并，$O(m+n)$次执行$S12'$，还有$O(m+n)$次另外的集合运算。如果我们使用Fischer在文献[14.37]和Hopcroft与Ullman在文献[14.38]中所描述的算法来完成不相交集合的并和S12'，那么从Tarjan在文献[14.39]所给的分析中得出可约简算法的复杂性为$O(m\alpha(m,n))$，这里$\alpha(m,n)$是增长很慢的函数，它与Ackermann函数$A(p,q)$的反函数有关，并定义如下：

$$\alpha(m,n)=\min\{z\geqslant 1 \mid A(z,4\lceil m/n\rceil)>\log_2 n\}$$

Ackermann函数的定义为

$$A(p,q)=\begin{cases} 2q, & p=0 \\ 0, & q=0 \text{ 且 } p\geqslant 1 \\ 2, & p\geqslant 1 \text{ 且 } q=1 \\ A(p-1,\ A(p,q-1)), & p\geqslant 1 \text{ 且 } q\geqslant 2 \end{cases} \tag{14.4}$$

注意：Ackermann函数是一个增长很快的函数。很容易发现：$A(3,4)$是一个很大的数，并且可以证明：如果$m\neq 0$且$\log_2 n<A(3,4)$，则有$\alpha(m,n)\leqslant 3$。上面提到的集合运算算法在文献[14.40]和[14.41]中也有描述。

算法14.8是非结构性的，即它没有给出一个用来压缩顶点从而简化一个可约的图的顺序。然而，如我们所要看到的那样，随着算法的展开，这种信息是很容易得到的。

在DFS期间，我们对各顶点按从n到1的顺序来配一个称为SNUMBER的数，并按照这个顺序完成对每个顶点的扫描。事实上，这与算法14.5的数组SCAN中相应元素置值为1的顺序相同。可以很容易地证明：

1. 如果(v,w)是树边，那么$\text{SNUMBER}(v)<\text{SNUMBER}(w)$，
2. 如果(v,w)是交叉边，那么$\text{SNUMBER}(v)<\text{SNUMBER}(w)$，
3. 如果(v,w)是后向边，那么$\text{SNUMBER}(v)>\text{SNUMBER}(w)$，
4. 如果(v,w)是前向边，那么$\text{SNUMBER}(v)<\text{SNUMBER}(w)$。

举例，在图14.21中我们在括号中表示了图14.8(a)所示图的相应顶点的SNUMBER数。

假设我们应用可约性算法，并且每次标记顶点v时，我们都把它与序对（$\text{HIGHPT1}(v)$，$\text{SNUMBER}(v)$）联系起来。当算法完成时，我们对顶点进行排序，并使得标记为(x_1, y_1)的一个顶点出现，在标记为(x_2,y_2)的顶点之前，当且仅当$x_1>x_2$或$x_1=x_2$且$y_1<y_2$。顶点的这一顺序称为约简顺序。注意：一个未标记的顶点v是与序对（0，$\text{SNUMBER}(v)$）相对应的。

假设v_a、v_b、v_c、…是可约简程序图G的约简顺序。令T为G的一个生成树，顶点s为根。用定理14.12和先前所描述的那些SNUMBER的性质，就很容易证明：树边(u, v_a)是指向v_a的唯一关联边。假设我们把v_a压入到u中。令G'为导出的可约简图，并令T'从T

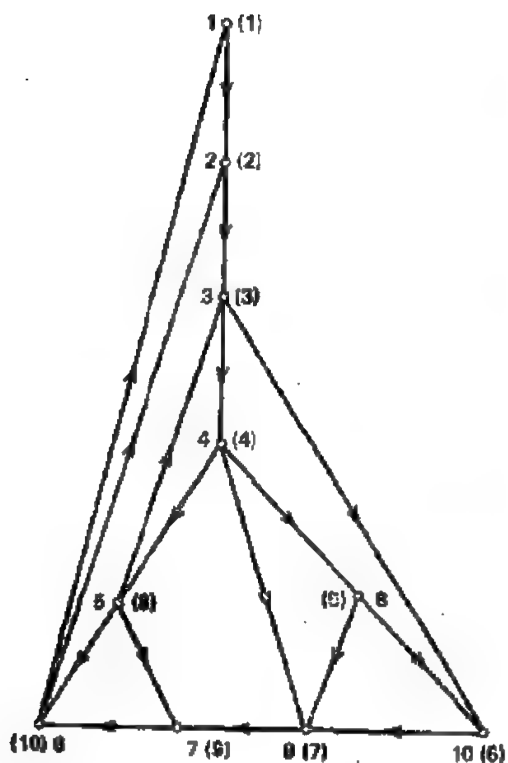


图 14.21　图14.18(a)的图，括号中为SNUMBER值

中为删去边（u，v_a）后所得到的树，那么T'显然是G'的一个DFS树。进而

1.G的一条交叉边或者什么也不对应，或者对应于G'的一条交叉边，或者对应于G'的前向边。

2.G的一条前向边或者什么也不对应，或者对应于G'的一条前向边。

3.G的一条后向边或者什么也不对应，或者对应于G'的一条后向边。

现在可以证明：G'中顶点的相关HIGHPT1值与在G中的相同。这点对于SNUMBER值也为真。因此，v_b、v_c、…也是G'的一个约简顺序。重复上述论证，我们得到下面定理。

定理14.13　如果一个程序图G是可约简的，那么我们就可以利用变换S_2（间或应用S_1）、按上述约简顺序来压入G的顶点。　□

例如，从图14.18(a)所给的HIGHPT1值以及图14.21所给的SNUMBER值都可以证明：对于图14.18(a)所示的图，序列4、5、3、8、10、9、7、6、2是一约简顺序。

14.6　程序图的控主

令G是以s为起始顶点的程序图。如果G中，顶点v在每条从s到w的有向路径上，那么v称为w的一个控主，并记为DOM(w)。如果v是w的一个控主，并且w的每个其它控主也控制v，那么v就称为w的一个直接控主，并记为IDOM(w)。例如，在图14.22(a)所示的程序图G中，顶点1是顶点9的直接控主。

可以证明：一个程序图$G=(V, E)$（除了起始顶点s外）有唯一的直接控主。边$\{(\text{IDOM}(w), w)\mid w\in V-\{s\}\}$形成以$s$为根的有向树，这棵树称为G的控主树，并使得$v$控制$w$，当且仅当$v$是控主树中$w$的一个正常前代顶点。如果$G$代表计算机程序中的一个控制流，那么控主树就提供了何种代码移动是安全的信息。图14.22(a)中程序图的控主树，如图14.22(b)所示。

我们现在推导一个求程序图控主树的算法，这个算法是由Lengauer和Tarjan在文献[14.42]中提出的。它是对早期Tarjan在文献[14.43]中提出的一种算法的更为简单迅速的改进。

令G为一程序图，其起始顶点为s。令T为G的一个DFS树。下面我们用DFS数来识别G的顶点。进而，记号$x\xrightarrow{*}y$表示了x是T中y的一个前代顶点，$x\xrightarrow{+}y$表示了在T中的

$x \xrightarrow{*} y$ 且 $x \neq y$，$x \to y$ 表示了 x 是 T 中 y 的父亲。

下面二个引理在算法推导时是很关键的。引理14.6很容易从定理14.7的论证中得出。利用这一结果，可以证明引理14.17。

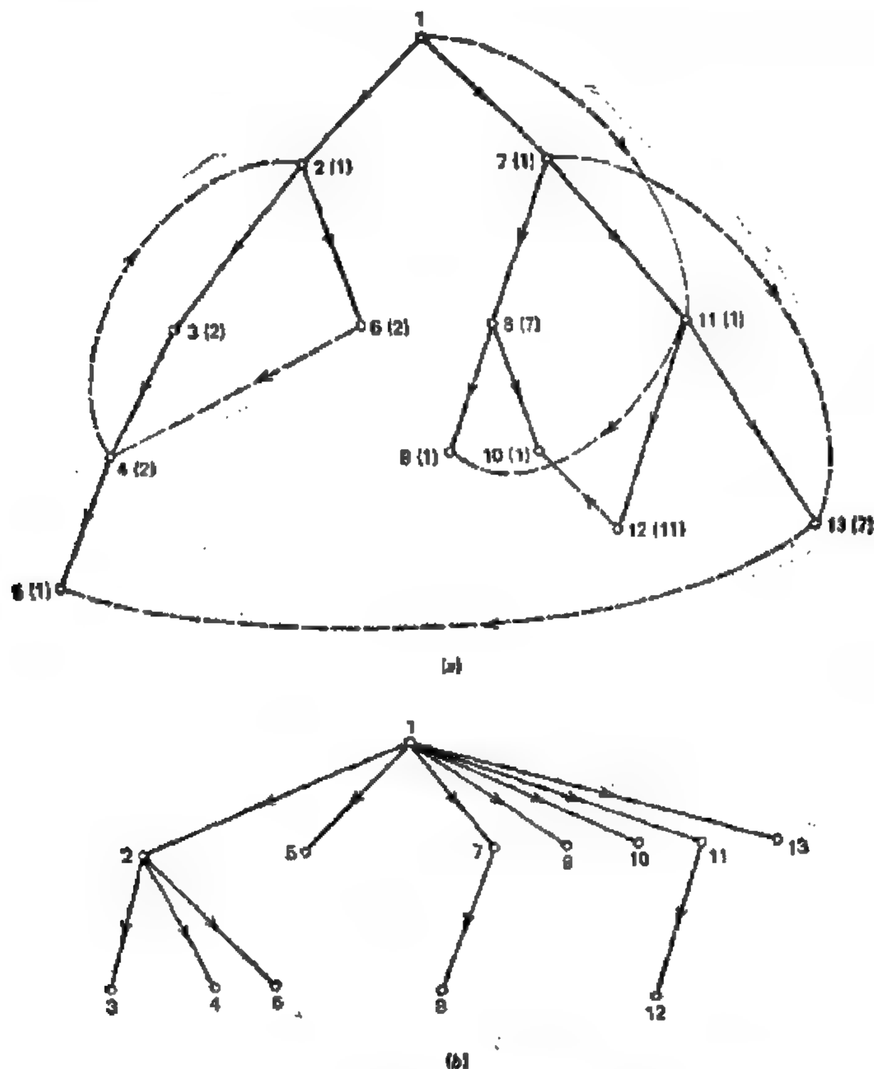


图 14.22 (a)一个程序图；(b)(a)图的控主树。

引理14.6 如果v和w是G的顶点，并满足$v<w$，那么任何从v到w的有向路径在T中必然包含v和w的一个公共前代顶点。 □

引理14.7 令$w \neq s$，$v \xrightarrow{*} w$，且P是从s到w的一条有向路径。令x为P中的最后顶点，并满足$x<v$，令y为P中继x以后的第一个顶点，并且满足$v \xrightarrow{*} y \xrightarrow{*} w$，如果Q：$x=v_0$，$v_1$，$v_2$，…，$v_k=y$是P中从$x$到$y$的那部分，那么对于$1 \leqslant i \leqslant k-1$，有$v_i>y$。 □

现在我们来对每个顶点$w \neq s$下定义，

$$\text{SDOM}(w)=\min\{v \mid 对 1 \leqslant i \leqslant k-1，存在一条有向路径 v=v_0, v_1, v_2, \cdots, v_k=w，并使得 v_i>w\}$$

SDOM(w)就称为w 的半控主。从上述的定义很容易得到

$$\mathrm{SDOM}(w) < w \tag{14.5}$$

在图14.22(a)中，实边和虚边分别表示属于一次DFS的树边和非树边。每个顶点的半控主表示在该顶点旁边的括号中。

作为第一步，Lengauer和 Tarjan 算法是计算所有顶点的半控主，然后用这些半控主来计算这些顶点的直接控主。

下面的定理提供了计算半控主的一种方法。

定理14.14 对于任一顶点$w \neq s$，

$$\mathrm{SDOM}(w)=\min(\{v \mid (v, w)\in E，且v<w\} \cup \{\mathrm{SDOM}(u) \mid u>w，且存在一条边(v, w)使得 u \xrightarrow{*} v\}) \tag{14.6}$$

证明

令x等于式(14.6)的右边。利用半控主的定义可以证明：$\mathrm{SDOM}(w) \leqslant x$。

为了证明$\mathrm{SDOM}(w) \geqslant x$，令$y=\mathrm{SDOM}(w)$，且令$y=v_0, v_1, \cdots, v_k=w$为一有向边，并对于$1 \leqslant i \leqslant k-1$，满足$v_i > w$。如果$k=1$，那么$(y, w) \in E$，且由(14.5)有$y<w$，因此，$\mathrm{SDOM}(w)=y \geqslant x$。假定$k>1$。令$j$是使$j \geqslant 1$和$v_j \xrightarrow{*} v_{k-1}$成立的最小数。因为$k-1$是$j$的一个候选者，所以这样的$j$存在。现在我们断言对$1 \leqslant i \leqslant j-1$，有$v_i > v_j$。

假如反设，对于某个$1 \leqslant i \leqslant j-1$，有$v_i < v_j$。那么取这个$i$，并使得$1 \leqslant i \leqslant j-1$且$v_i$是最小者。然后由引理14.6，有$v_i \xrightarrow{*} v_j$，这与$j$的选取相矛盾。这就证明了断言。

断言蕴含了$\mathrm{SDOM}(w)=y \geqslant \mathrm{SDOM}(v_j)$。由于$v_j > w$，$v_j \xrightarrow{*} v_{k-1}$，且$(v_{k-1}, w) \in E$，从式(14.6)得出$\mathrm{SDOM}(v_j) \geqslant x$，所以$\mathrm{SDOM}(w) \geqslant x$。这样，不论$k=1$或$k>1$，我们都有$\mathrm{SDOM}(w) \geqslant x$，这就证明了定理。 □

现在我们要求寻找一种从半控主来计算直接控主的方法。为了这个目标，我们如下进行。

下面三个引理很容易证明。引理14.9的证明要用引理14.6。

引理14.8 对于任一顶点$w \neq s$，$\mathrm{IDOM}(w) \xrightarrow{+} w$。

引理14.9 对于任一顶点$w \neq s$，令$v=\mathrm{SDOM}(w)$，则$v \xrightarrow{+} w$。

引理14.10 对于任一顶点$w \neq s$，令$v=\mathrm{SDOM}(w)$，则$\mathrm{IDOM}(w) \xrightarrow{*} v$。

引理14.11 令顶点v，w满足$v \xrightarrow{*} w$，那么或者$v \xrightarrow{*} \mathrm{IDOM}(w)$

或者$\mathrm{IDOM}(w) \xrightarrow{*} \mathrm{IDOM}(v)$

证明

令x为IDOM(v)的任一正常后继顶点，IDOM(v)也是v的一个正常前代顶点，那么存在一条从s到v的、不包含x的有向路径。通过连接这条路径与T中从v到w的通路，我们就得到一条从s到w的、不包含x的有向通路。从而IDOM(w)必须或者是v的一个

后继顶点或者是IDOM(v)的一个前代顶点。

利用前面的引理，我们接着来证明两个结果，从而提供一种从半控主来计算直接控主的方法。

定理14.5 令$w \neq s$，$v=\text{SDOM}(w)$。假设每一个使$v \overset{+}{\to} u \overset{*}{\to} w$成立的$u$都满足SDOM($u$)$\geqslant$SDOM($w$)，那么IDOM($w$)$=v$。

证明

由引理14.10，IDOM $(w)\overset{*}{\to} v$，因此，要证明IDOM(w)$=v$，只要证明v控制w就足够了。

考虑从v到w的任一有向路径P。令x为这条路径上使$x<v$的最后一个顶点。如果不存在这样的x，那么$v=s$就控制w。否则，令y为该路径上，继x之后而满足$v\overset{*}{\to} y \overset{*}{\to} w$的第一个顶点。令Q：$x=v_0$，$v_1$，$v_2$，…，$v_k=y$为P的从$x$到$y$的那部分。那么由引理14.7，对于$1\leqslant i\leqslant k-1$，有$v_i>y$。这与半控主定义结合，就使SDOM($y$)$\leqslant x<y=$SDOM($w$)。所以SDOM($y$)$<$SDOM($w$)。

根据定理的假设，对于每一个满足$v\overset{+}{\to} u\overset{*}{\to} w$的$v$，有SDOM($u$)$\geqslant$SDOM($w$)。这样，$y$就不可能是$v$的一个正常后继顶点。因为$y$满足$v\overset{*}{\to} y\overset{*}{\to} w$，所以就得出$y=v$，且$v$在P上。因为$P$的选择是任意的，所以$v$支配$w$。 □

定理14.16 令$w\neq s$，且$v=$SDOM(w)。令u为这样一个顶点，对于它，SDOM(u)是满足$v\overset{+}{\to} u\overset{*}{\to} w$的所有顶点$u$中的最小者。那么SDOM($u$)$\leqslant$SDOM($w$)且IDOM($u$)$=$IDOM($w$)。

证明

令z为使$v\to z\overset{*}{\to} w$成立的顶点，那么SDOM(u)$\leqslant$SDOM(z)$\leqslant v=$SDOM(w)。

由引理14.10，IDOM(w)是v的一个前代顶点，并且还是u的一个正常前代顶点。因此，由引理14.11，有IDOM(w)$\overset{*}{\to}$IDOM(u)。为了证明IDOM(u)$=$IDOM(w)，只要证明IDOM(u)控制w就够了。

考虑从s到w的任一路径P。令x为P中满足$x<$IDOM(u)的最后一个顶点。如果不存在这样的x，那么IDOM(u)$=s$就控制支配w。令y为P中继x之后、满足IDOM(u)$\overset{*}{\to} y\overset{*}{\to} w$的第一个顶点。就象在定理14.15的证明中那样，我们可以利用引理14.7来证明SDOM(y)$\leqslant x$。因为根据引理14.10，有IDOM(u)$\leqslant$SDOM(u)，所以我们有SDOM(y)$\leqslant x<$IDOM(u)$\leqslant$SDOM(u)。因此，SDOM(y)$<$SDOM(u)。

因为在从z到w的树路径上的顶点之中，u有一最小半控主，所以y不可能是v的一个正常后继顶点。进而，y也不可能既是IDOM(u)的正常后继顶点，又是u的前代顶点。如若真是这样情况，一条包含由s到SDOM(y)树路径的有向路径，对于$1\leqslant i\leqslant k-1$，先接在满足$v_i>y$的一条路径SDOM($y$)$=v_0$，$v_1$，$v_2$，…$v_k=y$之前，然后再接在从$y$到$u$的树路径所组成的有向通路之前，但应避开IDOM($u$)，但是没有从$s$到$u$的路径避开了IDOM($u$)。

剩下的唯一可能性就是IDOM(u)=y，从而IDOM(u)在从s到w的有向路径P上。因为P的选择是任意的，所以IDOM(u)控制w。 □

下面的主要结论是定理14.15和14.16的一个直接结果。

定理14.17 令$w \neq s$，且v=SDOM(w)。令u为一个顶点，对于它，SDOM(u)是满足$v \xrightarrow{+} u \xrightarrow{*} w$的所有顶点$u$中的最小者。那么

$$\text{IDOM}(w)=\begin{cases} v, & \text{若SDOM}(u)=\text{SDOM}(w) \\ \text{IDOM}(u), & \text{其它} \end{cases}$$ □

现在我们准备来描述 Lengauer 和 Tarjan 的控主算法。下面就是这个算法的主要步骤。

算法14.9 控主(Lengauer和Tarjan)

S1. 对给定的以起始顶点为根的程序图G=(V, E)进行一次DFS。

S2. 应用定理14.14来计算所有顶点的半控主。一个顶点一个顶点地按照它们DFS数的递减顺序进行计算。

S3. 利用定理14.17，就隐含地定义了每个顶点的直接控主。

S4. 明显地定义每个顶点的直接控主，按它们的DFS数的递增顺序逐个顶点进行计算。 □

S1步的实现是很简单的。下面我们用S1步中所分配的DFS数来表示顶点。

在描述S2、S3和S4步时，我们使用了数组FATHER(14.3节所定义的)、SEMI、BUCKET和DOM，其定义如下：

SEMI(w)

1. 在计算w的半控主前，

$$\text{SEMI}(w)=w$$

2. 在计算w的半控主后，

$$\text{SEMI}(w)=\text{SDOM}(w)$$

BUCKET(w) 这是一个顶点集合，其中顶点的半控主是w。

DOM(w)

1. S3步后，如果w的半控主是它的直接控主，那么DOM(w)就是w的直接控主。否则，DOM(w)就是使得$v<w$的顶点v，并且v的直接控主也是w的直接控主者。

2. S4步后，DOM(w) 就是w的直接控主者。

在进行了S1步后，算法同时进行S2和S3步，按顶点的DFS数的递减顺序来处理顶点$w \neq 1$。在计算过程中，算法保持了一个G的DFS树中所含有的林。这个林是由顶点集V和边集{(FATHER(w), w)|顶点w已被处理}组成。这个算法使用一个方法来建立这个林，并且使用另一个方法从中抽取信息。这些过程是：

LINK(v, w) 将边(v, w)加到林中。

EVAL(v)

1. 如果v是林中一个树的根，那么EVAL(v)=v。

2. 否则，令r为包含v的林中那个树的根，并令u为一个顶点，对于这个顶点，SEMI(u)

是满足 $r \xrightarrow{+} u \xrightarrow{*} v$ 的顶点中的最小者，那么EVAL(v)=u。

为了处理一个顶点w，这个算法是应用定理14.14来计算w的半控主的。从而这个算法指定了。

$$\text{SEMI}(w)=\min\{\text{SEMI}(\text{EVAL}(v))\mid(v,\ w)\in E\}$$

在计算之后，SEMI(w)就是w的半支控主者。这是从定理14.14和EVAL(v)的定义得出的。

在SEMI(w)计算之后，算法把w加到BUCKET(SEMI(w))中，并利用LINK(FATHER(w)，w)把一条新边加到这个林中。这就对w完成了S2步。

然后这个算法通过考虑BUCKET(FATHER(w))中的每个顶点来进行S3步。

令v为这样的一个顶点。利用定理14.17算法，隐含地计算出v的直接控主。令u=EVAL(v)。那么u就是满足 FATHER(w) $\xrightarrow{+} u \xrightarrow{*} v$ 的顶点，它的半控主者是最小的。如果SEMI(u)=SEMI(v)，那么FATHER(w)就是v的直接控主，并且算法指定了DOM(v)=FATHER(w)。否则，u和v具有相同的直接控主者，并且算法指定了DOM(v)=u。这就对v完成了S3步。

在S4步中，算法检验各顶点是否按它们的DFS数的递增顺序进行，并填写在S3步中没有明显计算出的直接控主。从而S4步如下进行：

对于每个i=2，3，…，n，如果DOM(i)≠SEMI(i)，那么置

$$\text{DOM}(i)=\text{DOM}(\text{DOM}(i))$$

作为控主算法的一个举例说明，考虑图14.22(a)所示的程序图。恰好在顶点11被处理之前，这个林示于图14.23(a)中。在这个阶段的SEMI数组的值表示在相应顶点旁边的括号内。现在我们来处理顶点11。

边（1，11）和（7，11）是指向11的关联边。所以

$$\text{SEMI}(11)=\min\{\text{SEMI}(\text{EVAL}(1)),\ \text{SEMI}(\text{EVAL}(7))\}$$

现在EVAL(1)=1，这是因为顶点1是林中的一个树根。由于同样原因，EVAL(7)=7，从而

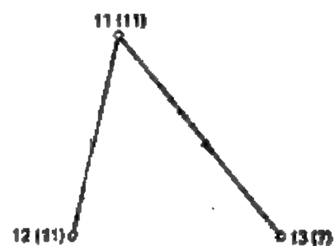
$$\begin{aligned}\text{SEMI}(11)&=\min\{\text{SEMI}(1),\ \text{SEMI}(7)\}\\&=\min\{1,\ 7\}\\&=1\end{aligned}$$

现在算法就把边（7，11）加到林中，把顶点11加到BUCKET（SEMI(11)）=BUCKET(1)中，这个新的林及其SEMI数组的值示于图14.23(b)。这样就对顶点11完成了S2步。

算法现在考虑BUCKET（FATHER(11)）=BUCKET(7)。顶点13是半控主等于7的唯一顶点，，所以BUCKET(7)={13}，现在EVAL(13)=11，这是因为SEMI(11)是满足 $7 \xrightarrow{+} u \xrightarrow{*} 13$（图14.23($b$)）的所有顶点$u$中的最小值。因为SEMI(13)≠SEMI(11)，算法就置DOM(13)=11。这样就对顶点11完成了S3步。

在对每个顶点w≠1进行S2和S3步后，所有顶点的半控主都是可求得的。在这个阶段，DOM数组和半控主的值给出如下：

顶　点	DOM	半支配者(SEMI)	顶　点	DOM	半支配者(SEMI)
2	1	1	8	7	7
3	2	2	9	1	1
4	2	2	10	1	1
5	1	1	11	1	1
6	2	2	12	11	11
7	1	1	13	11	7



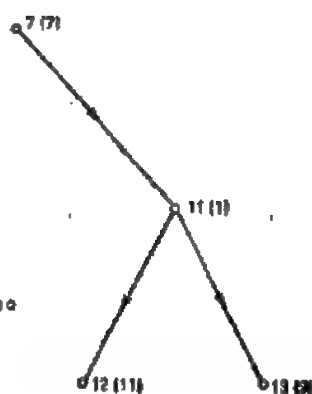


图 14.23

对于每个顶点 $w \neq 13$, $\mathrm{DOM}(w)=\mathrm{SEMI}(w)$。所以，对于除顶点13以外的所有顶点，$\mathrm{IDOM}(w)=\mathrm{DOM}(w)$。对于顶点13，我们计算

$$
\begin{aligned}
\mathrm{DOM}(13) &= \mathrm{DOM}(\mathrm{DOM}(13)) \\
&= \mathrm{DOM}(11)
\end{aligned}
$$

所以IDOM(13)=1。这就完成了算法的$S4$，并且得到示于图14.22(b)的控主树。

很显然，上述算法的复杂性关键取决于LINK和EVAL指令的实现，Tarjan在文献[14.44]中讨论了使用路径压缩的二种方法。其中之一介绍如下：

为了表示由LINK指令来建立起的林，算法使用了两个数组，ANCESTOR和LABEL。开始时，对于每个顶点v，ANCESTOR(v)=0及LABEL(u)=v。一般说，只有v是林中的一个树根，ANCESTOR(v)=0；否则，ANCESTOR(v)是林中v的前代顶点。

算法保持顶点的标记以便能满足下面的性质。令v为任一顶点，r为含有v的林中的树根，并令$v=v_k$，v_{k-1}，……，$v_0=r$为一个序列+。对于$1\leqslant i\leqslant k$，使得，ANCESTOR(v_i)=v_{i-1}。令x为这样一个顶点，其中使得SEMI(x)为顶点$x\in\{\text{LABEL}(v_i)\mid 1\leqslant i\leqslant k\}$中的最小值。那么我们有如下的性质。

x是这样一个顶点，其中使得SEMI(x)是满足

$$r\xrightarrow{+}x\xrightarrow{*}v$$

的顶点x中的最小者。为了进行LINK(v, w)，算法指定ANCESTOR(w)=v。为了进行EVAL(v)，算法顺着前代顶点指针，并且决定了序列$v=v_k$，v_{k-1}，…，$v_0=r$，从而对于$1\leqslant i\leqslant k$，有ANCESTOR(v_i)=v_{i-1}。如果$v=r$，那么算法置EVAL(v)=v。否则，对于$2\leqslant i\leqslant k$，算法置ANCESTOR(v_i)=r，同时按下法对顶点更新标号（保持前面所提到的性质）：

如果SEMI（LABEL(v_{i-1})）<SEMI（LABEL(v_i)），

那么LABEL(v_i)=LABEL(v_{i-1})。

然后算法置EVAL(v)=LABEL(v)。

Tarjan在文献[14.44]中已经证明：使用上面所描述的方法来实现（$n-1$）次LINK和（$m+n-1$）次EVAL的复杂性为$O(m\log n)$。如果我们使用更复杂的LINK和EVAL指令的实现方法，这在文献[14.44]中有所描述，那么算法就要求O（$m\alpha(m,n)$）次，这里$\alpha(m,n)$是上节所定义Ackermann函数的反函数。

对于其它的控主算法，可参阅文献[14.45]和[14.46]。

14.7 进一步阅读

文献[14.47]、[14.48]，[14.40]、[14.49]、[14.50]、[14.51]、[14.52]、和[14.53]是图论与组合算法中一些很好的参考书。讨论图的计算机表示的各种方法，可查阅文献[14.40]和[14.41]。

本书讨论的所有算法的运算时间都是以一个图的顶点数和边数的某个多项式为上限的。令P为可以用多项式时间算法解算的一类问题。还有大量问题被认为是不存在已知多项式时间算法的。其中有许多可以通过非确定性算法用多项式时间来解算，NP表示所有这类问题。一个问题是NP-难度的，如果对它解法所用的一个确定性的多项式时间算法

+原文中未说明序列。——译者注

可以用来寻求NP类中每个问题的一个确定性的多项式时间算法。在NP类中一个NP-难度问题称为NP-完全问题。从而，如果对NP-完全类中的任何问题找到一个确定性的多项式时间算法，那么这样的算法对NP类中的每个问题都存在。在Cook的一篇首创性论文[14.54]中，证明了可满足性问题是NP-完全的。Karp在文献[14.55]中已经证明，有大量问题是NP-完全的。Garey和Johnson在文献[14.56]与Even在文献[14.53]中给出了NP-完全性理论的一个精美的导引以及几个图论问题的NP-完全性证明。这也可参阅文献[14.40]、[14.51]和[14.57]。

在不少文献中介绍了许多有趣的图的算法，其中某些将在下章末尾列出。

14.8 习 题

14.1 有向图$G=(V,E)$的一个传递约简定义为一个任意图$G'=(V',E')$，其中G'具有尽可能少的边，并使得G'的传递闭包与G的传递闭包相等。设计一个用来求有向图的传递约简的算法。这个算法与传递闭包算法的关系怎样？（见文献[14.58]）

14.2 利用DFS，对下列问题设计算法：

(a) 一个图顶点的拓扑分类，

(b) 寻求一个图的桥，

(c) 求一个生成林，

(d) 求一个基本回路集合和一个基本割集集，

(e) 检测一个图是否是2-分的，

(f) 检测边的一个子集是否是一个图的割集。

14.3 考虑一个图G，且令T为G的一个DFS生成林。令ND(v)为v的后继顶点数（包括它本身）。证明：一个顶点w是v的一个后继顶点，当且仅当

$$\mathrm{DFN}(v)\leqslant\mathrm{DFN}(w)<\mathrm{DFN}(v)+\mathrm{ND}(v)$$

14.4 修改DFS算法，以包含ND(v)的计算。

14.5 令T为连通图G的一个DFS树，又令G_s为G的一个完全子图。试证：G_s的所有顶点都在T中的一条有向通路上。

14.6 令T为有向图G的一个DFS树。如果C是G中一条有向回路，以及v是C上具有最小深度优先数的顶点。试证：v是C上每个顶点在T中的前代顶点。

14.7 广度优先搜索（BFS）按如下的算法寻求一连通图G：

S1.开始时，G的所有顶点都未标记。

S2.任选顶点S，并标记为0。

S3.置$i=0$。

S4.令S为至少与一个标号为i的顶点邻接的、所有未标记的顶点集合。

S5.如果S为空集，则停机。否则，以$i+1$来标记S中的所有顶点。

S6.置$i=i+1$并转到S4。

试证：标记顶点时所经过的边构成了G的一个生成树。

14.8 证明：如何利用BFS算法来计算连通图G中从顶点s到图中所有顶点间的距离。（所谓顶点u和v之间的距离，我们指具有最少边数的$u-v$路径的长度。

14.9 证明：一个可约简程序图的后向边集合是唯一的，即所有DFS的生成林都具有相同的后向边

集合。（见文献[14.35]）

14.10 利用定理6.10的结果来设计一个求连通图的所有生成林的算法。

14.11 令T为具有n个顶点的一个树。令$\{1,2,\cdots\cdots,n\}$为T的顶点集合。我们可以把唯一的序列$(t_1,t_2,\cdots\cdots t_{n-2})$按下法与$T$联系起来：令$s_1$为$T$中度数1的第1个顶点，然后把$s_1$的邻接顶点记为$t_1$。现在从$T$中删去$s_1$。如果$s_2$是$T-s_1$中度数为1的第一个顶点，那么把$T-s_1$中$s_2$的邻接顶点记为$t_2$。删去$s_2$并重复这一操作，直到$t_{n-2}$被定义为止。序列$(t_1,t_2,\cdots\cdots,t_{n-2})$称为与T相关的Prüfer序列。显然，两个相异树具有相异的Prüfer序列。

给定一个序列$(t_1,t_2,\cdots\cdots,t_{n-2})$，其中每个$t_i\in\{1,2,\cdots\cdots,n\}$。设计一个算法用来构造出一个树$T$，其中$(t_1,t_2,\cdots\cdots,t_{n-2})$是Prüfer序列。

（可以从集合$\{1,2,\cdots\cdots,n\}$构造的$(n-2)$-字母序列数为n^{n-2}。每个这种序列都是K_n的一个生成树的Prüfer序列。所以在利用集合$\{1,2,\cdots\cdots,n\}$的任意$n-2$个字母（不需要是相异的）所构造的序列与K_n的生成树之间存在着一一对应关系。因此，K_n生成树的数目为n^{n-2}。这个证明起因于Prüfer的文献[14.59]。）

14.12 利用定理9.10的结果，设计求图的色数的一个算法。

14.9 参考文献

14.1 D. Gries, *Compiler Construction for Digital Computers*, Wiley, New York, 1971.

14.2 S. Warshall, "A Theorem on Boolean Matrices," *J. ACM*, Vol 9, 11—12(1962).

14.3 H. S. Warren, "A Modification of Warshall's Algorithm for the Transitive Closure of Binary Relations," *Comm. ACM*, Vol. 18, 218—220(1975).

14.4 V. L. Arlazarov, E. A. Dinic, M. A. Kronrod, and I. A. Faradzev, "On Economical Construction of the Transitive Closure of a Directed Graph," *Soviet Math. Dokl.*, Vol 11, 1209—1210(1970).

14.5 V. Strassen, "Gaussian Elimination is Not Optimal," *Numerische Math.*, Vol. 13, 354—356(1969).

14.6 M. J. Fischer and A. R. Meyer, "Boolean Matrix Multiplication and Transitive Closure," *Conf. Record, IEEE 12th Annual Symp. on Switching and Automata Theory*, 1971, pp. 129—131.

14.7 M. E. Furman, "Application of a Method of Fast Multiplication of Matrices in the Problem of Finding the Transitive Closure of a Graph," *Soviet Math. Dokl.*, Vol 11, 1252(1970).

14.8 I. Munro, "Efficient Determination of the Transitive Closure of a Directed Graph," *Information Proccessing Lett.*, Vol. 1, 56—58(1971).

14.9 P. E. O'Neil and E. J. O'Neil, "A Fast Expected Time Algorithm for Boolean Matrix Multiplication and Transitive Closure," *Inform. and Control*, Vol. 22, 132—138(1973).

14.10 J. Eve and R. Kurki-Suonio, "On Computing the Transitive Closure of a Relation," *Acta Informatica*, Vol. 8, 303—314(1977).

14.11 P. Purdom, "A Transitive Closure Algorithm," *BIT*, Vol. 10, 76—94(1970).

14.12 C. P. Schnorr, "An Algorithm for Transitive Closure with Linear Expected

Time," *SIAM J. Computing*, Vol. 7, 127—133(1978).

14.13 M. M. Syslo and J. Dzikiewicz, "Computational Exqerience with Some Transitive Closure Algorithms," *Computing*, Vol. 15,33—38(1975).

14.14 A. Pnueli, A. Lemqel, and S. Even, "Transitive Orientation of Graphs and Identification of Permutation Graphs,"*Canad. J. Math.*, Vol. 23, 160—175(1971).

14.15 P. C. Gilmore and A. J. Hoffman, "A Characterization of Comparability Graphs and of Interval Graphs," *Canad. J. Math.*, Vol. 16, 539—548(1964).

14.16 S. Even, A. Pnueli, and A. Lempel, "Permutation Graphs and Transitive Graphs," *J. ACM*, Vol. 19, 400—410(1972).

14.17 S. Even, *Algorithmic Combinatorics*, Macmillan, New York, 1973

14.18 C. L. Liu, *Introduction to Combinctorial Mathematics*, McGraw-Hill, New York, 1968.

14.19 R. E. Tarjan,"Depth-First Search and Linear Graph Algorithms," *SIAM J Computing*, Vol. 1, 146—160(1972).

14.20 J. Hopcroft and R. Tarjan, "Efficient Algorithms for Graph Manipulation," *Comm. ACM*, Vol. 16, 372—378(1973).

14.21 A. V. Aho, J. E. Hopcroft, and J. D. Ullman, "On Finding the Least Common Ancestors in Trees," *SIAMJ Computing*, Vol. 5, 115—132(1976).

14.22 J. Cocke, "Global Common Subexperssion Elimination," *SIGPLAN Notices*, Vol. 5, 20—24(1970).

14.23 J. D. Ullman, "Fast Algorithms for the Elimination of Common Subexpressions," *Acta Informatica*, Vol. 2, 191—213(1973).

14.24 K. Kennedy, "A Global Flow Analysis Algorithm," *Int. J. Computer Math.*, Vol. 3, 5—16(1971).

14.25 M. Schaefer, *A Mathematical Theory of Global Progrom Optimization*, Prentice-Hall, Englewood Cliffs, N. J., 1973.

14.26 A. V.Aho and J. D. Ullman, *The Theory of Parsing, Translation and Compiling, Vol. II—Compiling*, Prentice-Hall, Englewood Cliffs N. J., 1973.

14.27 F. E. Allen, "Control Flow Analysis," *SIGPLAN Notices*, Vol 5, 1—19(1970).

14.28 F. E. Allen, *Program Optimization, Annual Review in Automatic Programming*, Vol. 5, Pergamon, New York, 1969.

14.29 M. S. Hecht, *Flow Analysis of Computer Programs*, Elsevier, New York, 1977.

14.30 M. S. Hecht and J. D.Ullman, "Flow Graph Reducibility," *SIAM J. Computing*, Vol. 1, 188—202(1972).

14.31 J. Cocke and R. E. Miller, "Some Analysis Techniques for Optimizing Computer Programs," *Proc. 2nd Int. Conf. on System Sciences*, Honolulu, Hawaii 1969.

14.32 J. E. Hopcroft and J. D. Ullman, "An $n\log n$ Algorithm for Detecting Reducible Graphs," *Proc. 6th Annual Princeton Conf. on Information Sciences and Systems*, Princeton, N. J., 1972, pp. 119—122.

14.33 R. E. Tarjan, "Testing Flow Graph Reducibility," *Proc. 5th Annual ACM Symp. on Theory of Computing*, 1973, pp. 96—107.

14.34 R. E. Tarjan, "Testing Flow Graph Reducibility," *J. Comput. Sys. Sci.*, Vol. 9, 355—365(1974).

14.35 M. S. Hecht and J. D. Ullman, "Characterizations of Reducible Flow Graphs," *J. ACM*, Vol. 21, 367—375(1974).

14.36 J. M. Adams, J. M. Phelan, and R. H. Stark, "A Note on the Hecht-Ullman Characterization of Non-Reducible Flow Graphs," *SIAM J. Computing*, Vol. 3, 222—223(1974).

14.37 M. Fischer, "Efficiency of Equivalence Algorithms," in *Complexity of Computer Computations*(R. E. Miller and J. W. Thatcher, Eds.), Plenum Press, New York, 1972, pp. 153—168.

14.38 J. E. Hopcroft and J. D. Ullman, "Set Merging Algorithms," *SIAM J. Computing*, Vol, 2, 294—303(1973).

14.39 R. E. Tarjan, "On the Efficiency of a Good but Not Linear Set Union Algorithm," *J. ACM*, Vol. 22, 215—225(1975).

14.40 A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, Mass., 1974.

14.41 E. Horowitz and S. Sahni, *Fundamentals of Data Structures*, Computer Science Press, Potomac, Md., 1976.

14.42 T. Lengauer and R. E. Tarjan, "A Fast Algorithm for Finding Dominators in a Flow Graph," *Trans. on Prog. Lang. and Sys.*, Vol. 1, 121—141(1979).

14.43 R. E. Tarjan, "Finding Dominators in Directed Graphs," *SIAM J. Computing*, Vol. 3, 62—89(1974).

14.44 R. E. Tarjan, "Applications of Path Compression on Balanced Trees," *J. ACM*, Vol. 26, 690—715(1979).

14.45 A. V. Aho and J. D. Ullman, *Principles of Compiler Design*, Addison-Wesley, Reading, Mass., 1977.

14.46 P. W. Purdom and E. F. Moore, "Algorithm 430: Immediate Predominators in a Directed Graph," *Comm. ACM*, Vol. 15, 777—778(1972).

14.47 H. Frank and I. T. Frisch, *Communication, Transmission and Transportation Networks*, Addison-Wesley, Reading, Mass., 1971.

14.48 D. E. Knuth, *The Art of Computer Programming, Vol. 3: Sorting and Searching*, Addison-Wesley, Reading, Mass., 1973.

14.49 N. Christofides, *Graph Theory: An Algorithmic Approach*, Academic Press, New York, 1975.

14.50 E. L. Lawler, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York, 1976.

14.51 E. M. Reingold, J. Nievergelt, and N. Deo, *Combinatorial Algorithms: Theory and Practice*, Prentice Hall, Englewood Cliffs, N. J., 1977.

14.52 E. Minieka, *Optimization Algorithms for Networks and Graphs*, Marcel Dekker, New York, 1978.

14.53 S. Even, *Graph Algorithms*, Computer Science Press, Potomac, Md., 1979.

14.54 S. A. Cook, "The Complexity of Theorem Proving Procedures," *Proc. 3rd ACM Symp. on Theory of Computing*, 1971, pp. 151—158.

14.55 R. M. Karp, "Reducibility among Combinatorial Problems," in *Complexity of Computer Computations*(R. E. Miller and J. W. Thatcher, Eds.) Plenum Press, New York, 1972, pp. 85—104.

14.56 M. R. Garey and D. S. Johnson, *Computers and Intractability, A Guide to the Theory of NP-Completeness*, Freeman, San Francisco, Ca., 1979.

14.57 E.Horowitz and S. Sahni, *Fundamentals of Computer Algorithms*, Computer Science Press, Potomac, Md., 1978.

14.58 A. V. Aho, M. R. Garey, and J. D. Ullman, "The Transitive Reduction of a Directed Graph," *SIAM J. Computing*, Vol. 1, 131—137(1972).

14.59 H. Prüfer, "Neuer Beweis eines Satzes über Permutationen," *Arch. Math. Phys.*, Vol. 27, 742—744(1918).

第十五章　算法优化

本章，我们将研究几个与图的优化问题有关的算法。这些问题都是图论在运筹学和计算机科学的许多应用中产生的。下面讨论关于几个主要专题的算法。

1.最短路径，

2.最优树，

3.图的匹配，

4.网络流，

5.最优分支。

15.1 最短路径

设G是一有向连通图，其中每一条有向边与一个称做边的长度的正实数相关联。从顶点i到顶点j有向边的长度记为$w(i,j)$。如果从顶点i到顶点j不存在有向边，那么$w(i,j)=\infty$。在图G中，一条有向路径的长度为此路径中各边长度之和，一条最小长度的$s-t$有向路径称做从s到t的最短路径。最短$s-t$有向路径的长度称做从s到t的距离，记为$d(s,t)$。显然，对所有i，$d(i,i)=0$

本节，我们讨论下面两个问题：

1.找出从G中给定点s到其它所有顶点的最短路径。

2.找出G中所有有序顶点对间的最短路径。

这两个问题出现在许多优化问题中。比如，求传输网络的最小费用流，就包括了寻找一条从网络中源点到汇点的最短路径（见文献[15.1]）。

15.1.1 从图中给定点s到其它所有顶点的最短路径

现在介绍Dijkstra在文献[15.2]中给出的算法，它可以找出n-顶点有向连通图中从给定顶点s到其它所有顶点间的最短路径。下面的思路构成了Dijkstra算法的基础。

设V表示G的顶点集合，S是V的子集，并使$s\in S$。$\overline{S}$表示V中S的补集，于是$\overline{S}=V-S$。

在s到$\overline{S}$中顶点的所有有向路径中，令P是最短长度的路径。P的长度就叫做从s到$\overline{S}$的距离$d(s,\overline{S})$。令P：$s,\cdots,u,v$。那么很显然，$u\in S,v\in\overline{S}$。进而，P的$s-u$部分必定是所有顶点均在S中的最短$s-u$路径。于是

$$d(s,\overline{S})=d(s,u)+w(u,v)$$

由此，距离$d(s,\overline{S})$可用公式

$$d(s,\overline{S})=\min_{\substack{u\in S\\ v\in\overline{S}}}\{d(s,u)+w(u,v)\} \tag{15.1}$$

来计算。

如果 v 是 $\overline{S}$ 中的顶点。若对某个 $u\in S$，使得 $d(s,\overline{S})=d(s,u)+w(u,v)$，那么很显然

$$d(s,v)=d(s,\overline{S}) \qquad (15.2)$$

Dijkstra算法构造了 V 的一个子集序列 $S_0=\{s\}$，S_1，$S_2\cdots$，并满足下列条件：

1.如果 $s=u_0$，u_1，$\cdots$，u_{n-1} 都是 V 的顶点，并使得 $d(s,u_1)\leqslant d(s,u_2)\leqslant d(s,u_3)\leqslant\cdots\leqslant d(s,u_{n-1})$，那么对于 $i>0$，$S_i=\{s,u_1,u_2\cdots u_i\}$。

2.集合 S_i 一旦确定，从 s 到 u_1，u_2，$\cdots u_i$ 的最短路就将已知。

如果按上述来定义集合 S_i，那么由式（15.2），则有

$$d(s,u_{i+1})=d(s,\overline{S_i}) \qquad (15.3)$$

于是，由 S_i 来确定 S_{i+1}，就包括了计算 $d(s,\overline{S_i})$。

子集 S_1，S_2，$\cdots S_{n-1}$ 可按如下构造：由式（15.1）得，

$$d(s,\overline{S}_0)=\min_{\substack{u\in S_0\\ v\in\overline{S}_0}}\{d(s,u)+w(u,v)\}=\min_{v\in\overline{S}_0}\{w(s,v)\}$$

于是，由式（15.3）得，u_1 是具有性质

$$d(s,u_1)=\min_{v\in\overline{S}_0}\{w(s,v)\} \qquad (15.4)$$

的顶点。

如果 P_i 表示从 s 到顶点 u_i 的最短路径，那么很显然 $P_1:s,u_1$。

假定子集 S_0，S_1，$\cdots S_i$ 和路径 P_1，P_2，$\cdots P_i$ 已经确定。现在就来确定 S_{i+1}，首先用式（15.1）来计算 $d(s,S_i)$。由式（15.3）得，u_{i+1} 是 $\overline{S}_i$ 中具有性质

$$d(s,u_{i+1})=d(s,\overline{S}_i)$$

的顶点。由式（15.1），则存在顶点 $j\in S_i$，并使得

$$d(s,u_{i+1})=d(s,u_j)+w(u_j,u_{i+1})$$

所以，只要把边 (u_j,u_{i+1}) 接到路径 P_j 上，就可得到 P_{i+1}。

如果我们只是对从给定点 s 到给定点 t 的最短路径感兴趣，那么在上述步骤中，当确定了第一个含有 t 的子集 S_i 后，即可停止。

很显然，上述步骤中的每一步都需要计算式（15.1）的最小值。如果计算最小值要取每一条边，那么从 S_{i-1} 确定 S_i，就需要 $(i-1)\times(n-i)$ 次加法和 $\{i(n-1)-1\}$ 次比较。因此，整个算法的运算次数是

$$\sum_{i=1}^{n-1}\{(2i-1)(n-i)-1\}$$

结果总复杂度是 $O(n^3)$。然而，这样加法和比较中有许多是不必要的重复。

Dijkstra在他的算法中，从上一步到下一步，通过存贮计算信息来避免那些重复的加法和比较。这只需要通过标号程序就可办到，正如我们将要看到的那样，他把算法的复杂度改进到 $O(n^2)$。下面的思路可得出 Dijkstra 标号程序。假定我们进行标号，并使得对于 $i=1$，2，$\cdots$，每个顶点 v 的标号 $l_i(v)$ 都满足：

1.$l_0(s)=0$，$l_0(v)=\infty$，　　对于所有 $v\neq s$。

2. 对于 $i \geq 1$,

$$l_i(v)=d(s,v), \qquad \text{对于所有}v\in S_{i-1}$$

$$l_i(v)=\min_{u\in S_{i-1}}\{d(s,u)+w(u,v)\}, \qquad \text{对于所有}u\in \overline{S}_{i-1},$$

那么很显然，u_i是具有性质

$$d(s,u_i)=d(s,\overline{S}_{i-1})=\min_{v\in \overline{S}_{i-1}}\{l_i(v)\}$$

的顶点。

现在按照下面的方法，由$l_i(v)$来计算$l_{i+1}(v)$：

1. 对于$v\in S_i$， $l_{i+1}(v)=l_i(v)=d(s,v)$
2. 对于$v\in \overline{S}_i$

$$\begin{aligned}l_{i+1}(v)&=\min_{u\in S_i}\{d(s,u)+w(u,v)\}\\&=\min\{l_i(v),d(s,u_i)+w(u_i,v)\}\\&=\min\{l_i(v),l_i(u_i)+w(u_i,v)\}\end{aligned} \tag{15.5}$$

那么选择u_{i+1}，以使

$$d(s,u_{i+1})=d(s,\overline{S})=\min_{v\in \overline{S}_i}\{l_{i+1}(v)\} \tag{15.6}$$

注意：集合S_i确定之后，u_i的标号就不会再改变。

因此，对于所有$v\neq s$，Dijkstra算法从标号$l_0(s)=0$和$l_0(v)=\infty$开始，算法执行过程中，标号根据式（15.5）进行修改，标号$l_{n+1}(v)$给出从s到v的距离。

很显然，确定了u_{i+1}，就包括对于所有$v\in \overline{S}_i$来计算$l_{i+1}(v)$，然后再找出这些标号中的最小者，对于$i\geq 1$，前面式（15.5）的计算需要$(n-i-1)$次加法和$(n-i-1)$次比较，而后面式（15.6）的计算需要$\{(n-i)-2\}$次比较，因此，Dijkstra算法的复杂度显然是$O(n^2)$。

下面介绍Dijkstra算法的步骤，在这一步骤中，LABEL是存放顶点现时标号的数组，当一个顶点对某个i被置为u_i时，它就成为了永久标号，用数组PERM表示那些永久标号的顶点。如果PERM$(v)=1$，那么v就是永久标号顶点，注意：在这种情形下，v的标号等于$d(s,v)$，下面我们从PERM$(s)=1$、PERM$(v)=0$（所有$v\neq s$）开始。

PRED是记录顶点从哪个顶点得到永久标号的数组，如果v被永久标号，那么

$$v,\text{PRED}(v),\text{PRED}(\text{PRED}(v)),\cdots,s$$

是最短$s-t$有向路径中的顶点。

算法15.1 最短路径（Dijkstra）

S0. G是给定的有向图，其边上有长度，求出G中从顶点S到所有其他顶点的最短路径。

S1.（开始）令LABEL$(s)=0$，PERM$(s)=1$，PRED$(s)=s$，对于所有$v\neq s$，令LABEL$(v)=\infty$，PERM$(v)=0$，PRED$(v)=v$，

S2. 令$i=0$，$u=s$（u是最新被永久标号顶点，现在它就是s。）

S3.（计算LABEL(v)及更新PRED数组的元素），令$i=i+1$，对每一个尚未得到永久标号的顶点v进行下列操作：

1. 令$M=\min\{\text{LABEL}(v),\ \text{LABEL}(u)+w(u,v)\}$，

2. 如果$M<\text{LABEL}(v)$，那么令$\text{LABEL}(v)=M$和$\text{PRED}(v)=u$，

S4.（判明顶点u_i）在所有尚未得到永久标号的顶点中，找出具有最小标号的顶点w（如出现相等的情况，可任意选择），令 PERM(w) = 1和$u=w$，（$u_i=w$，是最新得到永久标号的），

S5. 如果$i<n-1$，那么转回S3，否则停机。（所有的最短路径均已找到，顶点标号给出了最短路径的长度。而v，PRED(v)，PRED(PRED(v))，…，s是一条最短$s-t$有向路径的顶点。）□

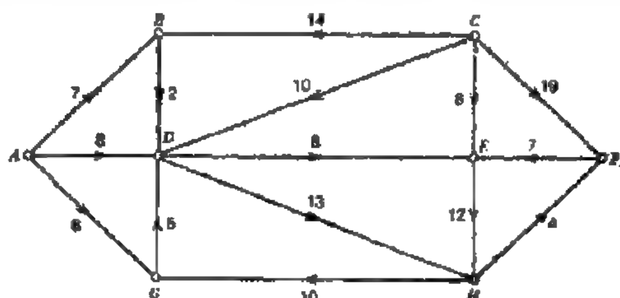


图 15.1

i	顶点							
	A	B	C	D	E	F	G	H
0	⓪*	∞	∞	∞	∞	∞	∞	∞
1	⓪	7	∞	8	∞	∞	⑥*	∞
2	⓪	⑦*	∞	8	∞	∞	⑥	∞
3	⓪	⑦	∞	⑧*	∞	∞	⑥	∞
4	⓪	⑦	∞	⑧	⑯*	∞	⑥	21
5	⓪	⑦	∞	⑧	⑯	∞	⑥	㉑*
6	⓪	⑦	∞	⑧	⑯	㉕*	⑥	㉑
7	⓪	⑦	(∞)*	⑧	⑯	㉕	⑥	㉑

(a)

PRED(A) = A　　PRED(E) = D

………………　　………………

………………　　………………

………………　　………………

从	到	最 短 路
A	B	A，B
A	C	无 路
A	D	A，D
⋮	⋮	⋮

(b)

图 15.2 Dijkstra算法示例。LABEL数组如(a)所示。

注意：在计算机程序中，∞是用尽可能大的数来表示。进而，如果顶点 v 的最终标号等于∞，那么就意味着从 s 到 v 不存在有向路径。

为了说明 Dijkstra 算法，参看图15.1所示图 G，其中一条边的长度在这条边的旁边标出。图15.2示出了LABEL和PRED两数组的部分元素值。

对于任意 i，LABEL数组中带圆圈的元素对应于永久标号顶点，即 S_i 中的顶点。带 * 号的元素对应于最新永久标号顶点，即顶点 u_i。从 PRED 和 LABEL数组元素的终值可以得到从 s 出发的最短路径和相应的距离。

直到现在，我们在讨论中假定所有长度都是非负的。如果某些长度是负的，那么Dijkstra算法则无效（为什么？）。因而，对此种算法做些修改，就可适用于不含负长度有向回路的一般网络。文献［15.3］已证明在最坏情况下，这个改进算法的复杂度是 $O(n\cdot 2^n)$ 而不是 $O(n^3)$，正如文献[15.4]所提出的那样。

有时我们希望求得第 2-(第3-，等等)最短路。这个问题及有关的问题在文献[15.5]、[15.6]，[15.7]，[15.8]，[15.9]，[15.10]等中做了讨论。对稀疏图设计的算法，可见文献[15.11]和[15.12]。

15.1.2 所有顶点对间的最短路径

假定我们希望在具有 n 个顶点的有向图中，找出所有 $n(n-1)$ 个有序顶点对间的最短路径。求得这些路径一个直观方法，就是运用 n 次 Dijkstra 算法。但是，在计算上有比这种更为有效的算法。这些算法甚至对于长度是负值、但不含负长度有向回路的情况也适用。现在讨论其中的一个。由文献[15.13]给出的这种算法，是以 Warshall 计算传递闭包的步骤（算法14.11）为基础的。

给定一个带有边长的 n 个顶点有向图 G。G 的顶点记为 1，2，… n。假定 G 中不存在负长度有向回路。令 $W=[w_{ij}]$ 是 G 中有向长度的 $n\times n$ 矩阵，即 $w(i,\ j)$ 是 G 中有向边 $(i,\ j)$ 的长度。如果不存在从 i 到 j 的有向边 $(i,\ j)$，那么置 $w_{ij}=\infty$。同时，对于所有的 i，置 $w_{ii}=0$。

Floyd算法从矩阵 $W^{(0)}=W$ 开始，构造一个 $n\times n$ 矩阵序列 $W^{(1)}$，$W^{(2)}$，…，$W^{(n)}$，$W^{(n)}$，并使得 $W^{(n)}$ 的元素 $w_{ij}^{(n)}$ 给出了 G 中从 i 到 j 的距离。根据下述原则：

$$w_{ij}^{(k)}=\min\{w_{ij}^{(k-1)},\ w_{ik}^{(k-1)}+w_{kj}^{(k-1)}\} \qquad (15.7)$$

就可从 $W^{(k-1)}=[w_{ij}^{(k-1)}]$ 构造出 $W^{(k)}=[w_{ij}^{(k)}]$。设 $P_{ij}^{(k)}$ 是所有那些仅以集合 $\{1,\ 2,\ \cdots k\}$ 中元素作为内部顶点的有向 $i-j$ 路径中长度最小的一条。下面的定理证明了 Floyol 算法的正确性。

定理15.1 对于 $0\leqslant k\leqslant n$，$w_{ij}^{(k)}$ 等于 $P_{ij}^{(k)}$ 的长度。

证明

证明可按定理14.1相同的线索进行。

通常，除了最短长度之外，我们还对具有这些长度的路径感兴趣。回忆一下，我们在Dijkstra算法中用PRED数组来记录出现在最短路径中的顶点。在Floyd算法中的实现方法如下所述。

在构造序列$W^{(0)}$，$W^{(1)}$，…，$W^{(n)}$的同时，还构造了另一个矩阵序列$Z^{(0)}$，$Z^{(1)}$，…，$Z^{(n)}$，并使得$Z^{(k)}$的元素$z_{ij}^{(k)}$给出了在$P_{ij}^{(k)}$中紧接着顶点 i 的顶点。显然有

$$z_{ij}^{(0)}=\begin{cases} j, & 若w_{ij}\neq\infty \\ 0, & 若w_{ij}=\infty \end{cases} \tag{15.8}$$

假定$Z^{(k-1)}=[z_{ij}^{(k-1)}]$，则根据下述规则得到$Z^{(k)}=[z_{ij}^{(k)}]$，令

$$M=\min\{w_{ij}^{(k-1)},\ w_{ik}^{(k-1)}+w_{kj}^{(k-1)}\}$$

那么

$$z_{ij}^{(k)}=\begin{cases} z_{ij}^{(k-1)}, & 若M=w_{ij}^{(k-1)} \\ z_{ik}^{(k-1)}, & 若M<w_{ij}^{(k-1)} \end{cases} \tag{15.9}$$

这条规则与算法15.1节$S3$中给出的更新PRED数组的规则很相似。其证明如下。如果

$$M=w_{ij}^{(k-1)}$$

那么

$$P_{ij}^{(k)}的长度=P_{ij}^{(k-1)}的长度$$

于是，$z_{ij}^{(k)}$与$z_{ij}^{(k-1)}$相同。

另一方面，如果$M<w_{ij}^{(k-1)}$，那么$P_{ij}^{(k-1)}$就是路径$P_{ik}^{(k-1)}$和$P_{kj}^{(k-1)}$的顺序连接。所以

$$z_{ij}^{(k)}=z_{ik}^{(k-1)}.$$

清楚可见，最短$i-j$路径是由顶点序列i，i_1，i_2，…，i_p，j给出的，其中

$$i_1=z_{ij}^{(n)},\ i_2=z_{i_1j}^{(n)},\ i_3=z_{i_2j}^{(n)},\ \cdots j=z_{i_pj}^{(n)} \tag{15.10}$$

注意：式（15.17）中，如果$w_{ik}^{(k-1)}$或$w_{kj}^{(k-1)}$等于∞，则$w_{ij}^{(k)}=w_{ij}^{(k-1)}$。这个直观结果将在下面描述Floyd算法时用到。在这个算法中，我们还增加了对负长度有向回路的检测。

算法15.2　所有顶点对间的最短路径（Floyd）

S1　$W=[w_{ij}]$是给定有向图G中的$n\times n$有向长度矩阵。这里对所有 $i=1,2,\cdots,n$，$w_{ii}=0$。$Z=[z_{ij}]$是$n\times n$矩阵，其中

$$z_{ij}=\begin{cases} j, & 若w_{ij}\neq\infty \\ 0, & 若w_{ij}=\infty \end{cases}$$

S2　置$k=0$。

S3　置$k=k+1$。对所有使$w_{ik}\neq\infty$的$i\neq k$以及所有使$w_{kj}\neq\infty$的$j\neq k$,进行下列操作：

1.置$M=\min\{w_{ij},\ w_{ik}+w_{kj}\}$。

2.如果$M<w_{ij}$，那么置$z_{ij}=z_{ik}$且$w_{ij}=M$。

S4　1.如果任意$w_{ii}<0$，那么顶点i就是在某个负长度有向回路中，则停机。

2.如果每个$w_{ii}\geq0$、且$k=n$，那么$[w_{ij}]$就给出了所有最短路径的长度，并且z_{ij}给出了最短$i-j$有向路径中、顶点i后面的第一个顶点，停机。

3.如果所有$w_{ii}\geq0$，但$k<n$，那么除去S3。 □

Floyd算法是目前求全部最短路径的最为有效的算法，它也同样适用于求负长度有向网络中的最短路径。至于求全部最短路径的其它有效算法，可参见文献[15.14]。

文献[15.15]也提出了一个求全部最短路径的算法。其后不久，文献[15.16]指出了文

献[15.15]中的一个错误，并介绍了改正的方法。

文献[15.17]给出了一个求解最短路径算法，以及相关问题的详尽索引。

15.2 最小加权路径长度树

一个m-元树是每个顶点的出度至多为m的有向树。曾记得，有向树根的入度是零，而其它顶点的入度是1，通常，m-元树是把根画在顶部，所有边的指向朝下，所有与根有相同距离的顶点都排在同一水平线上。

设T是一个m-元树，$M=\{0, 1, 2, \cdots m-1\}$是具有$m$个数字的字母表。假定给$T$中的每条边都算定集合$M$中的一个数字，并使得从同一顶点射出的两条边不会算定相同的数字。因此，T的每个顶点都与一个字相关联，这个字是把从根到达比顶点所经过边的给定的数字串联起来而得到的。这样与T的叶相关联的字叫做码字，称它们构成了前缀码。

例如，在图15.3所示3-元树的情况下，码字是00，010，120，121，22，20，21。

在前缀码中，每个字的一个有趣而又有用的性质就是没有一个字是其他字的开头。

假定我们已经把L个信息$M_1M_2, \cdots M_L$编成前缀码的字。如果把某些经过编码的信息所组成的一个序列在信道中传输，那么在接收端就得到一个由传输信息所对应的码字串接而成的一序列数字。为了从这个序列中复原信息，需要把比序列分解成前缀码中的字。把序列分解成码字的过程叫做解码，并且利用对应于前缀码的树，这一过程就极易做到。

例如，考虑一个由图15.3的树所对应的前缀码字而串接成的序列120202200。为了对这个序列进行解码我们从左到右搜索序列中的数字。在搜索过程中，我们从根开始，沿着对应搜索数字的边移动，直到叶为止。对应于这片叶的码字就是所给序列中的第一个字。于是，我们得到序列120202200中第一个字120。然后对余下的序列202200重复解码过程。从而分辨出序列120202200中第二，第三和第四个码字分别为20，22和00。

从上述解码过程显而易见，解出一个字的费用与这个字中数字的数目成正比。如果w_i是信息M_i出现的频率，那么预期解码费用就与$\sum_{i=1}^{L} w_i l_i$有关，这里l_i是由根到信息M_i所对应的叶的路径长度。和式$\sum_{i=1}^{L} w_i l_i$叫做树的加权路径长度。于是，通过选择码字的长度，

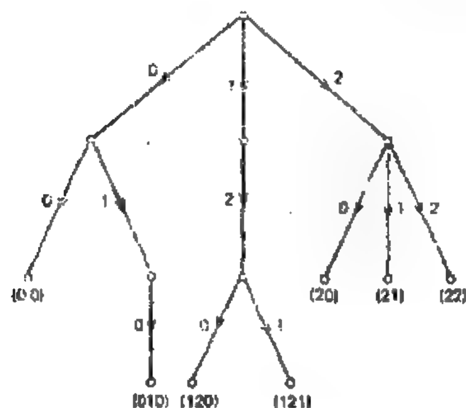


图 15.3 3-元树

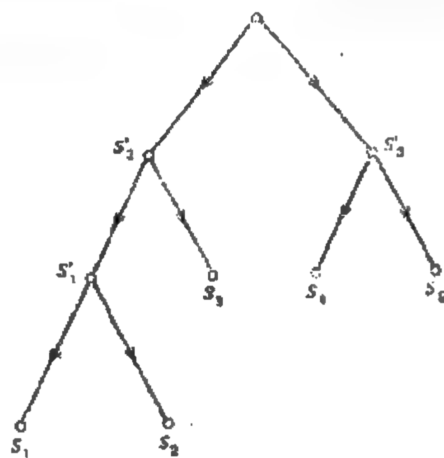


图 15.4

使得所形成的树具有最小加权路径长度，预期解码费用就可达到最小。这样就引出了对于给定一组权w_1，w_2，$\cdots w_L$，如何构造出具有最小加权路径长度的m-元树的问题。

考虑上述问题的解答之前，先讨论产生这个问题的另一种场合。

假定有L个表S_1，S_2，$\cdots S_L$，每个表都是由按递增序排列的整数组成。现在我们打算把这些集合弄成一个表，其元素仍按递增序排列。为此，首先合并其中任意两个（比方说，S_1和S_2），由此得到一个新表S'。接着再合并集合$\{S_1'S_3', S_4', \cdots, S_L'\}$中的任意两个，继续这个过程，直至得到一个单一表为止。二元树可用来描述这样合并的策略。

例如，考虑图15.4所示树。它描述了合并五个表S_1，S_2，S_3，S_4及S_5的一种方法。这里首先合并S_1和S_2，得到了S_1'。然后S_1'与S_3合并得到S_2'。下一步再合并S_4和S_5得到S_3'。最后，S_2'与S_3'合并就得到了所需的单一表。

一种合并任意两个表的方法所述如下。从两个给定表的每一个中，取出第一个元素即最小元素。把这两个元素中较小的一个，作为所需合并表中的一个元素。然后从合并表中移去这个元素。在相同的两个表上重复这种操作，其中有一个表比以前变短，直到它们合并成一个单一表为止。

很显然，合并任何两个表的费用与表中元素数目成正比。所以合并表S_1，S_2，$\cdots S_L$的费用等于$\sum_{i=1}^{L}|S_i|\,l_i$，这里l_i是涉及到表S_i中元素合并过程的次数。

例如，合并图15.4所示S_1，S_2，$\cdots S_5$的费用等于

$$3|S_1|+3|S_2|+2|S_3|+2|S_4|+2|S_5|$$

从图15.4可以看出，在这种情况下，l_i实际上等于从根到对应于S_i的叶的路径长度。于是，合并表S_1，S_2，$\cdots S_L$的费用最小问题等价于，对数$|S_1|$，$|S_2|$，$\cdots$，$|S_L|$，构造一个具有最小加权路径长度的二元树。

Huffman 在文献[15.18]中对上述问题给出了巧妙的解答。本节的余下部分致力于Huffman 算法的推导。考虑一个具有L片叶的m-元树T。设l_1，$l_2\cdots l_L$代表从T的根到叶的路径长度。这些l_i叫做对应于叶的路径长度，$\lambda=(l_1, l_2, \cdots l_L)$叫做$T$的路径长度矢量。$\lambda$的特征和$S(\lambda, m)$定义如下：

$$S(\lambda, m)=\sum_{i=1}^{L}m^{-l_i} \qquad (15.11)$$

首先建立$\lambda=(l_1, l_2\cdots l_L)$是$m$-元树的路径长度矢量充要条件。当然，假定每个$l_i$都是正整数。

引理15.1 m-元树的路径长度矢量$\alpha=(l_1, l_2, \cdots l_L)$的特征和小于或等于1。

证明

通过对λ中最大长度的归纳加以证明

如果λ中的最大长度等于1，那么所有l_i都等于1，很显然，在这种情况下，引理当然成立。令引理对最大长度小于k的每一个路径长度矢量均成立。假定λ中最大长度等于k在相应的m-元树T中，令v_1，v_2，$\cdots v_{d^+(r)}$是根r的子。注意，r的出度$d^+(r)$小于或等于m。令T_i表示以v_i为根的T的子树，则T_i中路径长度显然比它们在T中的相应部分少1。所以，如果S_i是T_i的路径长度矢量特征和，那么

$$S(\lambda,m)=\frac{1}{m}\left(S_1+S_2+\cdots+S_{d^+(r)}\right)$$

因为$d^+(r)\leqslant m$，且由归纳假设，对所有i，$S_i\leqslant 1$，则可推出$S(\lambda, m)\leqslant 1$。 □

引理15.2 矢量$\lambda=(l_1, l_2, \cdots, l_L)$是某一$m$-元树的路径长度矢量，如果它满足特征和条件，

$$\sum_{i=1}^{L} m^{-l_i}\leqslant 1 \tag{15.12}$$

这里l_i是正整数。

证明

排列长度l_i，以使$l_1\leqslant l_2\cdots\leqslant l_L$。设$\mu$等于$l_L$的长度，那么

$$l_{L-\mu}<l_{L-\mu+1}=l_{L-\mu+2}=\cdots=l_L$$

现在，以m^{l_L}乘（15.12）式，则得

$$\sum_{i=1}^{L-\mu} m^{l_L-l_i}+\mu\leqslant m^{l_L} \tag{15.13}$$

令$\mu=\sigma m+\rho$，这里σ和ρ都是非负的，且$0\leqslant\rho\leqslant m-1$，并定义$\pi$如下：

$$\pi=\begin{cases}0, & 若\rho=0\\ m-\rho, & 若\rho>0\end{cases}$$

那么可以把π加到式15.13左边，但不改变其不等号。于是

$$\sum_{i=1}^{L-\mu} m^{l_L-l_i}+\mu+\pi\leqslant m^{l_L} \tag{15.14}$$

注意：$\mu+\pi=\tau m$，这里

$$\tau=\begin{cases}\sigma, & 若\rho=0\\ \sigma+1, & 若\rho>0\end{cases} \tag{15.15}$$

现在，用m^{l_L}去除式（15.14），得

$$\sum_{i=1}^{L-\mu} m^{-l_i}+\tau m^{-(l_L-1)}\leqslant 1 \tag{15.16}$$

以上不等式左边的量是长度为$L-\mu+\tau$矢量的特征和，其中最大长度比给定矢量λ中最大长度少1。

现在，通过对最大长度大小的归纳来证明引理。

如果$l_L=1$，对于所有i，$l_i=1$，那么式（15.12）要有$L\leqslant m$。在这种情形下，因为矢量λ是由L条全部从根射出的边所构成的m-元树路径长度矢量，引理显然成立。

下面，假定引理对于所有最大长度小于k的矢量都成立。

令λ中的最大长度等于k。如前所述，我们可以构造一个具有最大长度等于$k-1$的$L-\mu+\tau$个长度的矢量λ'。在λ'中，至少有τ个长度等于$k-1$，而λ'的其余$L-\mu$个长度和λ的前$L-\mu$个长度相同。进而注意：$\tau=\sigma$或$\sigma+1$。

因为λ'满足特征和的条件式(15.16)，则由归纳假设可知，必然存在一个以λ'为其路径长度矢量的m-元树T'。考察T'的任意τ个长度为$k-1$的叶，如果我们把m条边与这些

顶点中任意σ个的每个顶点都相接，而其余每个顶点都与ρ条边相接，如可能，那么就得到一个$\mu=\sigma m+\beta$片叶、其路径长度等于k的新m-元树T。T的其余路径长度和λ的前$L-\mu$个路径长度相同。所以，对于T，λ就是路径长度矢量。 □

如果路径长度矢量$\lambda=(l_1, l_2, \cdots l_L)$使下面和式最小：

$$\sum_{i=1}^{L} w_i l_i$$

则称λ对矢量$W=(w_1, w_2, \cdots w_L)$是最优的。

如果λ对T是最优的，那么$l_i \geqslant l_j$就意味着$w_i \leqslant w_j$。所以可假定$l_1 \leqslant l_2 \leqslant \cdots \leqslant l_L$和$w_1 \geqslant w_2 \geqslant \cdots \geqslant w_L$。$m=1$的情形显然是成立的，在下面的讨论中，假定$m \geqslant 2$。下一个结论是用已知权矢量来构造最优路径长度矢量Huffman算法的基础。

定理15.2 设$\lambda=(l_1, l_2, \cdots l_L)$是对于已知权矢量$w=(w_1, w_2, \cdots w_L)$的最优路径长度矢量，假定$l_1 \leqslant l_2 \leqslant \cdots \leqslant l_L$，且$w_1 \geqslant w_2 \cdots \geqslant w_L$。那么$l_{L-d+1}=l_{L-d+2}=\cdots=l_L$，其中如果$m=2$，则$d=2$，如果$m>2$，则$d$由下式给出：

$$d=\begin{cases} m, & \text{若} L \equiv 1 \pmod{m-1} \\ m-1 & \text{若} L \equiv 0 \pmod{m-1} \\ \rho, & \text{若} L \equiv \rho \pmod{m-1} \text{及} 2 \leqslant \rho \leqslant m-2 \end{cases} \tag{15-17}$$

证明

定理指出，在最优路径长度矢量中，至少有d个长度等于最大长度。对$m=2$的情形，定理的证明就是个简单练习。所以，我们仅考虑$m>2$的情形。

从特征和条件可知

$$m^{l_L}-\sum_{i=1}^{L} m^{l_L-l_i} \geqslant 0 \tag{15.18}$$

现在证明

$$m-2 \geqslant m^{l_L}-\sum_{i=1}^{L} m^{l_L-l_i} \tag{15.19}$$

如果式(15.19)不成立，那么

$$m^{l_L}-\sum_{i=1}^{L} m^{l_L-l_i} \geqslant m-1$$

于是

$$\sum_{i=1}^{L-1} m^{l_L-l_i}+m \leqslant m^{l_L}$$

用m^{l_L}除上式两边，则得

$$\sum_{i=1}^{L-1} m^{-l_i}+m^{-(l_L-1)} \leqslant 1$$

但是，上式可λ是最优矢量相矛盾。因而，式(15.19)成立。

因为$m \equiv 1 \pmod{m-1}$，所以对于每个非负整数k，$m^k \equiv 1 \pmod{m-1}$。因而，

$$\begin{aligned} m^{l_L}-\sum_{i=1}^{L} m^{l_L-l_i} &\equiv 1-L \pmod{m-1} \\ &\equiv 1-\rho \pmod{m-1} \end{aligned} \tag{15.20}$$

这里$L=\rho(\bmod m-1)$。至此，

$$1-\rho(\bmod m-1)=\begin{cases}0, & 若\rho=1\\ 1, & 若\rho=0\\ m-\rho, & 若2\leqslant\rho\leqslant m-2\end{cases} \qquad (15.21)$$

由式(15.18)和(15.19)可知，$m^{l_L}-\sum_{i=1}^{L}m^{l_L-l_i}$ 是非负的，且小于 $m-1$。由此，从式(15.20)和(15.21)得到

$$m^{l_L}-\sum_{i=1}^{L}m^{l_L-l_i}=\begin{cases}0, & 若\rho=1\\ 1, & 若\rho=0\\ m-\rho, & 若2\leqslant\rho\leqslant m-2\end{cases} \qquad (15.22)$$

假定 j 是使$l_j<l_L$的最后一个下标，故有

$$l_1\leqslant l_2\leqslant\cdots\leqslant l_j\leqslant l_{j+1}=l_{j+2}=\cdots=l_L$$

现在用(15.22)式来证明$L-j\geqslant d$，以此来推出定理。式(15.22)可改写为

$$m^{l_L}-\sum_{i=1}^{j}m^{l_L-l_i}=\begin{cases}L-j, & 若\rho=1\\ L-j+1, & 若\rho=0\\ L-j+m-\rho, & 若2\leqslant\rho\leqslant m-2\end{cases}$$

因为上述方程的左端可被m除尽，所以有：

若$\rho=1$，则$L-j=km$，k 为某个正整数。

若$\rho=0$，则$L-j+1=km$，k 为某个正整数。

若$2\leqslant\rho\leqslant m-2$，则$L-j+m-\rho=km$，$k$ 为某个正整数。

于是，由式(15.17)，得

若$\rho=1$，　　　则$L-j\geqslant m=d$

若$\rho=0$，　　　则$L-j\geqslant m-1=d$

若$\rho 2\leqslant\rho\leqslant m-2$，则$L-j\geqslant\rho=d$ □

现在开始叙述对于给定的权矢量W_i来构造最优路径长度矢量的 Huffman算法。权矢量W_i中权的数目，记为L_i。如果$L_0\leqslant m$，那么所对应的路径长度矢量全部由1组成。因此，我们假定$L_0>m$。

Huffman 算法从W_0开始，构造出一个权矢量序列W_1，W_2，…，W_s，并使$L_0>L_1>\cdots>L_s=m$。对于每个L_i，d_i的定义同式(15-17)一样，即

$$d_i=2,\ 若m=2$$

否则，

$$d_i=\begin{cases}m, & 若L_i\equiv1\ (\bmod m-1)\\ m-1, & 若L_i\equiv0\ (\bmod m-1)\\ \rho, & 若L_i\equiv\rho\ (\bmod m-1)且2\leqslant\rho\leqslant m-2\end{cases} \qquad (15.23)$$

在由W_i来构造W_{i+1}中应用d_i。

因此，算法从下式开始

$$\lambda_s=(\overbrace{1,1,\cdots,1}^{m})$$

并且构造出一个路径长度矢量为 λ_s，λ_{s-1}，$\cdots\lambda_0$，而使 λ_i 是 W_i 的最优路径长度矢量的序列。假定在每个λ_i中，长度的排列顺序与所相应的W_i中权的排列顺序相同。

算法15.3　最优路径长度矢量（Huffman）

S1．W_0是给定的权矢量，置$i=0$。

S2．令$W_i=(w_1, w_2, \cdots w_{Li})$，且有$w_1\geqslant w_2\geqslant\cdots\geqslant w_{Li}$。$W_{i+1}$的构造如下

1.计算(15.23)式中的d_i。

2.计算$P_i=w_{Li-di+1}+w_{Li-di+2}+\cdots+w_{Li}$，即$P_i$是$W_i$中倒数$d_i$个权之和。

3.W_{i+1}由w_1，w_2，$\cdots w_{Li-di}$和P_i按递减顺序排列而成。

S3．置$i=i+1$。若$L_i=m$，转到S4。否则，返回S2。

S4．令

$$\lambda_i=(\overbrace{1,1,\cdots,1}^{m})$$

（注意现在$i=s$），置$i=i-1$。

S5．从λ_{i+1}构造λ_i如下：

1.对于W_i中前L_i-d_i个权的每一个，都标定一个等于它在W_{i+1}中数值的长度。

2.对于W_i中后d_i个权的每一个，都标定一个比W_{i+1}中P_i数值多1的长度。

S6．如果$i=0$，那么就停止；λ_0是W_0的最优路径长度矢量。否则，置$i=i-1$，转到S5。 □

很显然，λ_s是路径长度矢量。由(15.23)式，对所有i，$d_i\leqslant m$。算法中的构造过程保证了，对于所有i，λ_i都是路径长度矢量。

在图15.5中，用$W_0=(9,7,6,5,5,3,3,2,1)$和$m=3$举例说明Huffman算法。对应于路径长度矢量λ_3、λ_2、λ_1和λ_0的树T_3、T_2、T_1和T_0，如图15.6所示。（注意，T_i是通过把T_{i+1}中代表权P_i的叶与d_i条边连接起来构造的。）

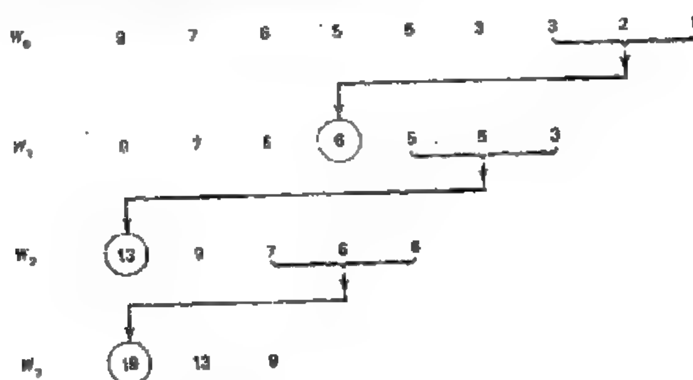


图 15.5　Huffman算法的举例说明

下面证明Huffman算法的正确性

定理15.3　设W_0，W_1，$\cdots W_s$，和λ_0，λ_1，$\cdots\lambda_s$是Huffman算法中所构造的权矢量和路径长度矢量。对于每个$0\leqslant i\leqslant s$，λ_i对W_i来说都是最优的。

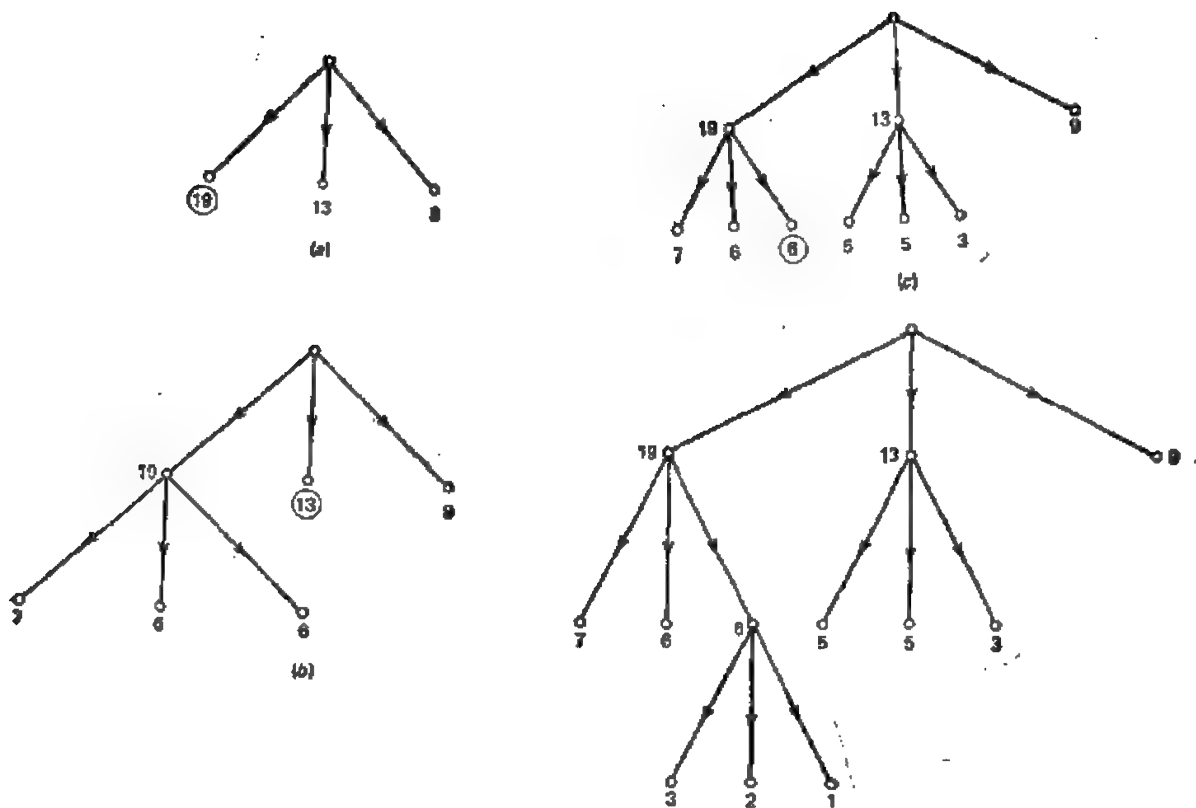


图 15.6 (a)树T_3; (b)树T_2; (c)树T_1; (d)树T_0.

证明

很显然，路径长度矢量

$$\lambda_s = (\overset{\longleftarrow m \longrightarrow}{1,1,\cdots,1})$$

对W_s来说是最优的. 现在证明：如果λ_{i+1}对W_{i+1}是最优的，那么λ_i对W_i也是最优的.

假设相反，即λ_{i+1}对W_{i+1}是最优的，但λ_i对W_i不是最优的. 那么令λ_i^*对W_i是最优路径长度矢量. 设

$$W_i = (p_1,\ p_2,\ \cdots,\ p_{L_i}),\ 且有p_1 \geqslant p_2 \geqslant \cdots \geqslant p_{L_i}$$

$$\lambda_i = (l_1,\ l_2,\ \cdots,\ l_{L_i})$$

$$W_{i+1} = (q_1,\ q_2,\ \cdots,\ q_{L_{i+1}}),\ 且有q_1 \geqslant q_2 \geqslant \cdots \geqslant q_{L_{i+1}}$$

$$\lambda_{i+1} = (t_1,\ t_2,\ \cdots,\ t_{L_{i+1}})$$

$$\lambda_i^* = (l_1^*,\ l_2^*,\ \cdots,\ l_{L_i}^*)$$

因为λ_i对W_i不是最优的，则有

$$\sum_{k=1}^{L_i} p_k l_k > \sum_{k=1}^{L_i} p_k l_k^* \tag{15.24}$$

从Huffman构造中的S5可知，对某一$1 \leqslant u \leqslant L_{i+1}$，

$$q_u = P_{L_i - s_i + 1} + P_{L_i - s_i + 2} + \cdots + P_{L_i}$$

（注意，q_u和P_i是相同的，其余的q_i是p_1，p_2，$\cdots p_{L_i-s_i}$）。

另外，$t_u=l_{L_i}-1$，所以

$$\sum_{k=1}^{L_{i+1}} q_k t_k = \sum_{k=1}^{L_i} (p_k l_k) - q_u \tag{15.25}$$

因为λ^*是最优矢量，根据定理15.2，则有

$$l^*_{L_i-s_{i+1}} = l^*_{L_i-s_{i+1}} = \cdots = l^*_{L_i}$$

如若用从λ_i到λ_{i+1}的相同形式的变换，那么从λ^*，就得到了一个路径长度矢量$\lambda^*_{i+1}=(t^*_1, t^*_2, \cdots, t^*_{L_{i+1}})$，并满足

$$\sum_{k=1}^{L_{i+1}} q_k t^*_k = \sum_{k=1}^{L_i} (p_k l^*_k) - q_u \tag{15.26}$$

现在，用式（15.24）、（15.25）和（15.26），可以证明

$$\sum_{k=1}^{L_{i+1}} q_k t^*_k < \sum_{k=1}^{L_{i+1}} q_k t_k$$

这就意味着λ_{i+1}不是最优矢量，这与假设相矛盾。 □

15.3 最优二元搜索树

有序树是一个有向树，有向树中每个顶点的子顶点都规定了一个顺序。我们画有序树，通常是把根放在上面，每个顶点的子顶点从左到右按给定顺序排列。对有序二元树来讲，以顶点v的左边子顶点为根的子树叫做v的左子树。v的右子树也同样定义。

设$A=\{a_1, a_2, \cdots, a_n\}$为一集合，其元素顺序为$a_1<a_2<\cdots<a_n$。集合$A$的二元搜索树是一棵有序二元树，其中每个顶点$v$都是用$A$中元素$l(v)$来标号的，并使得

1.对v的左子树中每个顶点u，$l(u)<l(v)$。

2.对v的右子树中每个顶点u，$l(u)>l(v)$。

3.对于A中的每个元素x，恰好存在一个顶点v，并使得$l(v)=x$。

假定A是泛集S的子集。令$B=\{b_0, b_1, \cdots, b_n\}$为一集合，并使得

1.$b_i(1\leqslant i\leqslant n-1)$代表所有元素$x\in S-A$的集合，其中$a_i<x<a_{i+1}$。

2.b_0代表所有元素$x\in S-A$的集合，其中$x<a_1$。

3.b_n代表所有元素$x\in S-A$的集中，其中$x>a_n$。

A的扩展二元搜索树是A的二元搜索树，其中有$n+1$片叶代表B中元素。注意，在扩展二元搜索树中，每个内部顶点的出度等于2。

例如，图15.7所示树是集合$A=\{a_1, a_2, a_3, a_4\}$上的两棵不同扩展二元搜索树。在扩展二元搜索树中，叶从左到右是按b_0，b_1，$\cdots b_n$的顺序出现。

下面我们把A的扩展二元搜索树简称为A的二元搜索树。

给定一个泛集S的子集A和A的二元搜索树T。集合S可以是英文字母上所有名字的集合，并按字母顺序排列。如果我们只是对确定$x\in S$是否属于A感兴趣，那么一种常用的方法就是把x与对应于T中根的元素来进行比较。这样会出现四种情形，分述之如下：

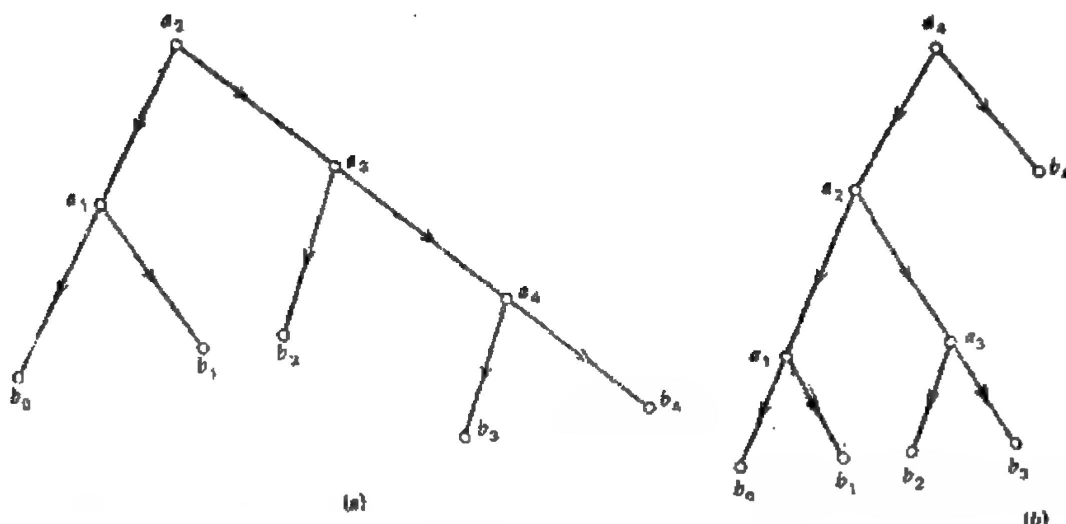


图 15.7 二元搜索树的例子

情形 1 不存在根（二元搜索树T是空的）。于是，x不在A中，搜索以失败告终。

情形 2 x与根上元素相同，于是搜索成功地终止。

情形 3 x小于根上元素，所以搜索继续沿根的左子树进行。

情形 4 x大于根上元素，所以搜索继续沿根的右子树进行。

很显然，成功的搜索终止在T的内部顶点上，不成功的搜索终止在T的叶上。

假定把从根到顶点v的路径长度定义为顶点v的深度，则可看到，当搜索成功地终止在T的内部顶点v之前所进行的比较次数等于v的深度加1。另一方面，如果搜索不成功地终止在某片叶上，那么所进行的比较次数就等于这片叶的深度。

令p_1，p_2，…p_n分别表示a_1，a_2，…，a_n的选取频率，q_0，q_1，…，q_n分别表示搜索终止在T中所代表b_0，b_1，…b_n的叶的频率。那么，T上平均搜索时间（对所有搜索）正比于树T的费用，其定义如下：

$$\sum_{i=1}^{n} p_i\,(a_i\text{的深度加 }1)+\sum_{j=0}^{n} q_j(b_j\text{的深度})$$

例如，如果$p_1=0.2$、$p_2=0.2$、$p_3=0.1$、$p_4=0.1$，$q_0=0.1$、$q_1=0.1$、$q_2=0.05$、$q_3=0.05$、$q_4=0.1$，那么图15.7(a)所示树的费用为

$$\{p_1(2)+p_2(1)+p_3(2)+p_4(3)\}$$
$$+\{q_0(2)+q_1(2)+q_2(2)+q_3(3)+q_4(3)\}=2.05$$

而图15.7(b)所示树的费用为

$$\{p_1(3)+p_2(2)+p_3(3)+p_4(1)\}$$
$$+\{q_0(3)+q_1(3)+q_2(3)+q_3(3)+q_4(1)\}$$
$$=2.4$$

上例提出了下面的问题。给定非负权p_i和q_i，对于集合$A=\{a_1, a_2, \cdots, a_n\}$来构造一个最小费用二元搜索树，$A$中的元素按$a_1<a_2<\cdots<a_n$次序排列。把这样一个树叫做对权$q_0$，$p_1$，$q_1$，$p_2$，…，$q_{n-1}$，$p_n$，$q_n$的最优二元搜索树，是很方便的。

现在讨论的构造最优二元搜索树的算法，就是基于这类树如下的一些有用性质。

设T是权为q_0，p_1，q_1，p_2，…，p_n，q_n的最优二元搜索树。如果a_k是T的根上的元素，那么T的根的左子树就是权为q_0，p_1，q_1，p_2，…，p_{k-1}，q_{k-1}的最优二元搜索树，T的根的右子树就是权为q_k，p_k，…，p_n，q_n的最优二元搜索树。

对于$0\leq i\leq j\leq n$，令T_{ij}表示对集合$\{q_i, p_{i+1}, q_{i+1}, \cdots, p_j, q_j\}$的最优二元搜索树，$r_{ij}$表示$T_{ij}$的根，$c_{ij}$表示$T_{ij}$的费用。$T_{ij}$的权$w_{ij}$被定义为等于$q_i+(p_{i+1}+q_{i+1})+\cdots+(p_j+q_j)$。$T_{ii}$是由叶$b_i$组成的树，所以$c_{ii}=0$，$w_{ii}=q_i$。

假定a_k是T_{ij}的根，$i<j$，那么如上观察，a_k的左子树$T_{i,k-1}$是对集合$\{q_i, p_{i+1}, \cdots, p_{k-1}, q_{k-1}\}$的最优二元搜索树。$a_k$的右子树$T_{kj}$是对集合$\{q_k, p_{k+1}, \cdots, p_j, q_j\}$的最优二元搜索树（见图15.8）。因为$T_{ij}$中顶点的深度比它在$T_{i,k-1}$或$T_{k,j}$中的深度多1，由此推得

$$
\begin{aligned}
c_{i,j} &= c_{i,k-1}+w_{i,k-1}+c_{kj}+w_{kj}+p_k \\
&= w_{ij}+c_{i,k-1}+c_{kj}
\end{aligned}
\qquad (15.27)
$$

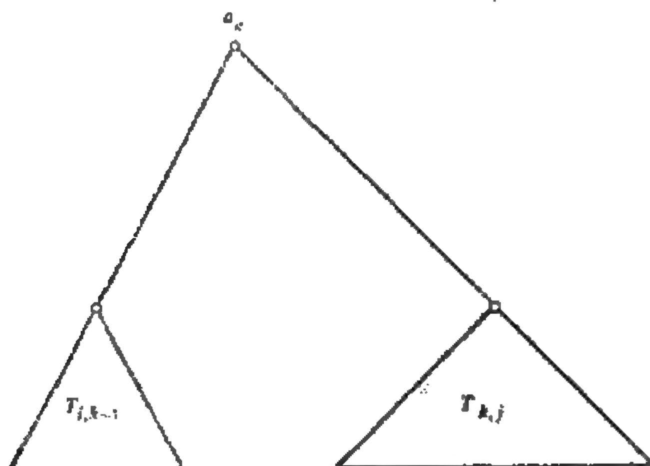


图 15.8 以a_k为根的子树T_{ij}

清楚可见，只要通过计算，使式（15.27）之和的最小k值，就能找到T_{ij}的根r_{ij}，$i<k\leq j$。这样就构成了算法15.4的基础，即计算r_{ij}和c_{ij}是按不断增加$j-i$值的顺序进行的。这个算法出自文献[15.9]。

算法15.4 最优二元搜索树（Gilbert及Moore）

S1 给定非负权p_1，p_2，…，p_n及q_0，q_1，…，q_n。对于$i=0, 1, 2, \cdots n$，令

$$w_{ii}=q_i,$$
$$c_{ii}=0,$$
$$r_{ii}=b_i.$$

S2 对于$l=1,2,\cdots,n$，进行S3。

S3 对于$i=0,1,2,\cdots,n-l$，进行S4。

S4 置

$$j=i+l,$$

$$w_{ij}=w_{i,j-1}+p_j+q_j.$$

令m是使和式$c_{i,k-1}+c_{kj}$为最小的k值，$i<k<j$。置

$$c_{ij}=w_{ij}+c_{i,m-1}+c_{mj},$$

$$r_{ij}=a_m.$$

S5 停机 □

用上述算法计算出一组r_{ij}值后，就可经过下面步骤来构造T_{0n}：

1. r_{0n}是T_{0n}的根。如果$r_{0n}=a_k$，那么a_k就以$r_{0,k-1}$为左子顶点，以r_{kn}为右子顶点。
2. 通常，考虑内部顶点a_m。如果$a_m=r_{ij}$，那么a_m就以$r_{i,m-1}$为左子顶点，以r_{mj}为右子顶点。

现举例说明算法15.4和上述构造最优二元搜索的步骤。

考虑四个元素$a_1<a_2<a_3<a_4$，且$p_1=2$，$p_2=1$，$p_3=3$ $p_4=1$，$q_0=1$，$q_1=2$，$q_2=3$，$q_3=2$，$q_4=1$。表15.1给出了用算法15.4所计算的w_{ij}、r_{ij}和c_{ij}的值。在表中，每行的元素值都是从左到右进行计算的。进而，任一行元素值只能在前面一行元素值计算出以后，才能进行计算。图15.9示出了所对应的最优二元搜索树。

表15.1

$l=j-i$ ↓ \ i→	0	1	2	3	4
0	$w_{00}=1$ $c_{00}=0$ $r_{00}=b_0$	$w_{11}=2$ $c_{11}=0$ $r_{11}=b_1$	$w_{22}=3$ $c_{22}=0$ $r_{22}=b_2$	$w_{33}=2$ $c_{33}=0$ $r_{33}=b_3$	$w_{44}=1$ $c_{44}=0$ $r_{44}=b_4$
1	$w_{01}=5$ $c_{01}=5$ $r_{01}=a_1$	$w_{12}=6$ $c_{12}=6$ $r_{12}=a_2$	$w_{23}=8$ $c_{23}=8$ $r_{23}=a_3$	$w_{34}=4$ $c_{34}=4$ $r_{34}=a_4$	
2	$w_{02}=9$ $c_{02}=14$ $r_{02}=a_2$	$w_{13}=11$ $c_{13}=17$ $r_{13}=a_3$	$w_{24}=10$ $c_{24}=14$ $r_{24}=a_3$		
3	$w_{03}=14$ $c_{03}=27$ $r_{03}=a_2$	$w_{14}=13$ $c_{14}=23$ $r_{14}=a_3$			
4	$w_{04}=16$ $c_{04}=34$ $r_{04}=a_3$				

很容易证明，算法15.4的复杂性为$O(n^3)$，从表中r_{ij}值表构造树过程的复杂性为$O(n)$。于是，最优二元搜索树可在$O(n^3)$时间内构造出。

Knuth在文献[15.20]已经证明了：T_{ij}的根位于$r_{i,j-1}$和$r_{i+1,j}$之间，即$r_{i,j-1}\leqslant r_{ij}\leqslant r_{i+1,j}$。所以，在S4中，就可以把对$m$的搜索限制在$r_{i,j-1}$到$r_{i+1,j}$的范围内。经过这样的修改，算法15.4的复杂性就变为$O(n^2)$。在文献[15.21]中，讨论了Knuth算法的几个推广。

文献[15.22]给出了在所有p_i值均等于零的特殊情况下来构造最优二元搜索树的$O(n^2)$算法，还可参见文献[15.23]。文献[15.24]证明这个算法可以在$O(n\cdot\log n)$时间内实现。文

献[15.25]对同样问题给出了另一个与文献[15.22]相近似的O($n\log n$)算法。这个算法易于证明，它是以变分法为基础的。

人们对于搜索方法和与最优搜索树有关的问题，进行了广泛的研究。对于这些专题的详细讨论和有关的参考资料，可参阅文献[15.24]和[15.26]，近期文章还有[15.27]。

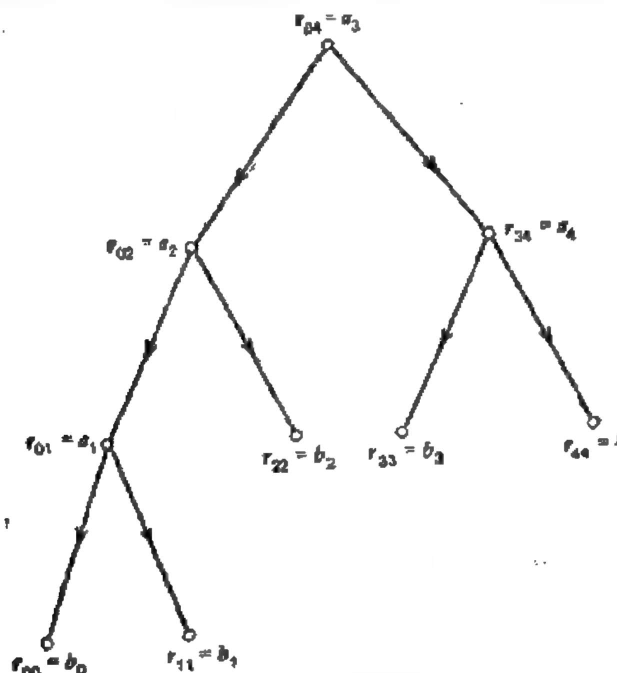


图 15.9 最优二元搜索树的构造

15.4 图中的最大匹配

本节将考虑构造图的最大匹配问题。首先讨论Edmond在文献[15.28]中提出的构造匹配的基本方法，然后叙述Gabow在文献[15.29]中的一个算法，它基本上是Edmonds方法的有效实现。

对二分图情形，一种更为有效的方法将在15.5节讨论。在15.6节将考虑有关人员最优分配和时间表安排的应用。

15.4.1 Edmond方法

Edmond算法是以Berge定理（定理8.20）为基础的，Berge定理指出某一匹配是最大的，当且仅当不存在相对于此匹配的增广路径。所以，给定了一个图和一个初始匹配M，就可按下述方法着手求得一个最大匹配。

找出一条相对于M的增广路径P，就得到比M多一条边的匹配$M\oplus P$。对于这个新的匹配，找出一条增广路径，处理方法同前。如此继续下去，直至得到一个不存在增广路径的匹配为止。因此，由Berge定理可知，这个匹配是最大的。

这样，这个问题基本上归结为，用有效方法来寻找一条相对于给定匹配的增广路径。本文中最重要的概念就是Edmond引入的“花”，现叙述如下：

为了找出相对于匹配M的增广路径，我们必须在一个非饱和顶点（比方说u）上开始

搜索。如果存在一条从u到u'的增广路径P（注意，u'也是非饱和顶点），那么在P中，u'或者与u相邻接，或是与一个饱和顶点v相邻接，而且从u到这样的顶点的距离是一个偶数，即从存在一条u到v为偶数长度的交错路径。这就意味着，搜索增广路径只需在一组可供选择的顶点集内进行，即从u到那些存在偶数长度的交错路径的顶点内进行。

例如，令v_1，v_2，…，v_r是与u相邻接的顶点（图15.10）。如果其中任何一个顶点是非饱和的，那么就找到了一条增广路径。否则，令u_1，u_2，…，u_r是它们各自在匹配M中的对偶。至此，可选顶点集是由顶点u，u_1，u_2，…，u_r组成，然后从可选顶点集中取出一个未检查的顶点，比如u_1。如果u_1有一个非饱和的相邻顶点，那么就找出了一条增广路径。

否则，假定u_1不与可选集中任何顶点相邻接。如果v_1'，v_2'…，v_s'是与u_1相邻接的那些顶点，对于所有i和j，使得$v_i \neq v_j$，那么它们的对偶u_1'，u_2'…，u_s'也加入可选顶点集中。

如果在搜索可选集中某个顶点时，发现此顶点与另一个已在可选集的顶点相邻接，那么这就产生了一个奇回路（具有奇数长度的回路）。这个具有奇数长度的闭合交错路径，

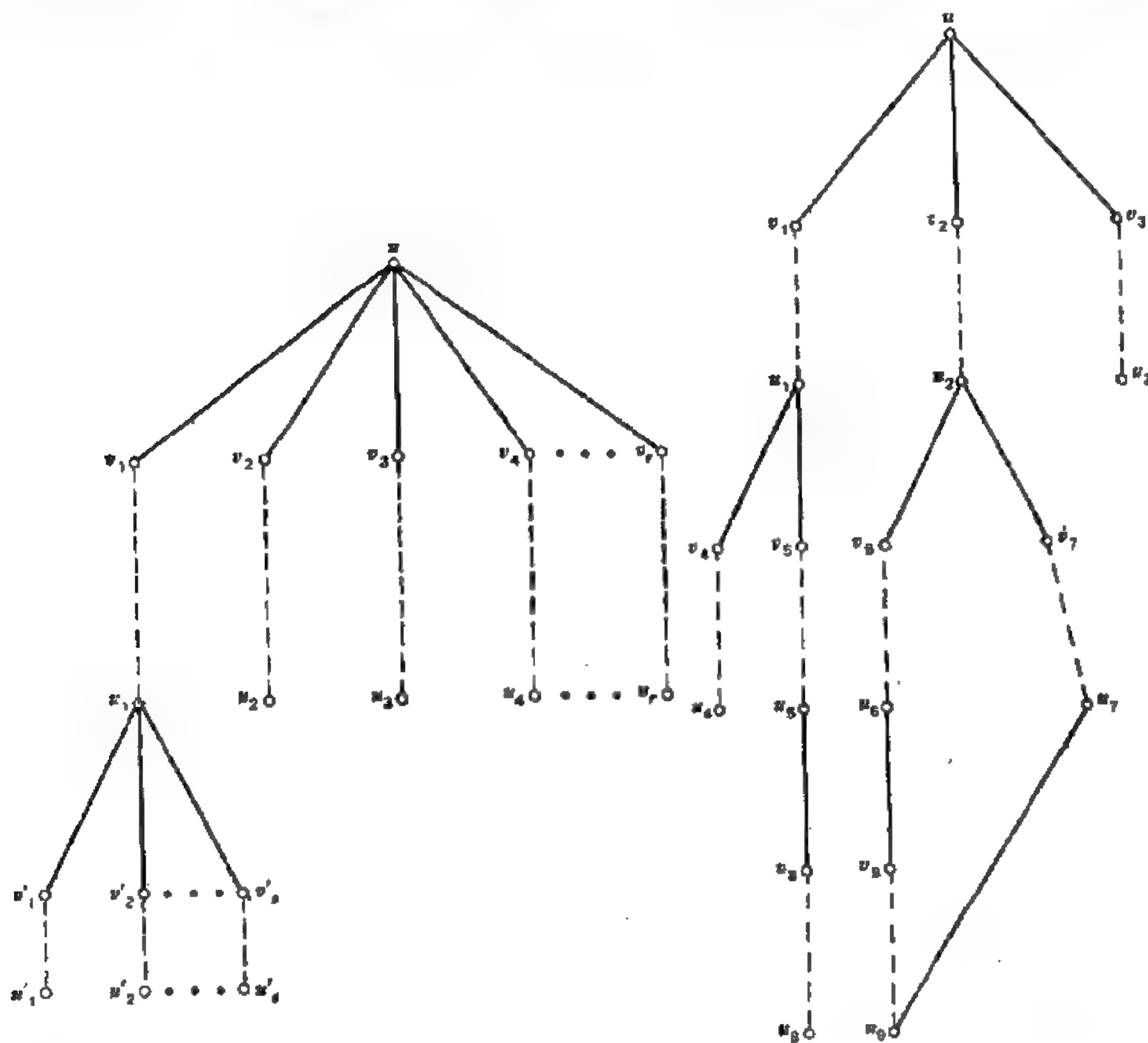


图 15.10

图 15.11 花的形成

叫做花。例如，在图15.11中加入边(u_9,u_7)就生成了花$(u_2,v_8,u_8,v_9,u_9,u_7,v_7,u_2)$。在加入这条边之前，可选集由$u$，$u_1$，$u_2$，…，$u_9$组成。但是，一旦生产了花，则顶点$v_8$，$v_7$和$v_9$也就加入了可选集，因为现在找出了从$u$到这些顶点的具有偶数长度的交错路径。例如，图15.11中，$(u,v_2,u_2,v_7,u_7,u_9,v_9)$是一条到达$v_9$的具有偶数长度的交错路径。所以，一旦生成了花，我们就发现：那朵花中的所有顶点都加入了可选集。

在Edmond算法中，一旦生成了花，就用一个单顶点来代替花中的所有顶点，这个新顶点叫做伪顶点。与花中一个或多个顶点相邻接的所有顶点现在都与这个新顶点相邻接。于是我们得到了一个缩减图。在这个缩减图上，继续寻找增广路径。当我们又找到一朵花时，就重复这个过程。

在这个方法中，如果最后在某个缩减图上找不到一个增广路径，那么就意味着：在原始图中，相对于当前匹配的增广路径也不存在。于是，当前匹配是最大的。

另一方面，如果在某个缩减图中找到一条增广路径，那么意味着：在原始图中也存在一条增广路径P。为了求出路径P，需要通过展开前面所得到的花来仔细回溯。

Edmond算法中所要求的花的收缩和展开，使这个算法的复杂性为$O(n^4)$，这里n是图中顶点的数目。

Gabow用一种有效的标号方法和合适的数组来记录相应花的结构，从而避免了花的收缩和展开。借助于此，算法复杂性可达到$O(n^3)$。Gabow所有的标号方法与文献[15.30]、[15.31]及[15.32]在匹配算法中所使用的方法相似。

15.4.2 Gabow方法

首先讨论Gabow算法的基本思路并定义Gabow算法中用到的不同数组。

令给定的图有n个顶点和m条边。此算法开始时，先给图的顶点和边编号。顶点的边号从1到n，边的编号为$n+2$，$n+4$，……，$n+2m$。边(x,y)的编号，记为$N(x,y)$。编号为0的虚拟顶点也要用到。

END是数组，其元素编号从$n+1$到$n+2m$。对于每条边，END中存在两个包含这条边端点的相邻元素。于是，如果边(v, w)编号为k，（其中，对于某个$1 \leqslant i \leqslant m$，$k=n+2i$），那么么$END(k-1)=v$，$END(k)=w$。所以，给定了边的编号，用这个数组易于确定它的端点。

Gabow算法从初始匹配开始来构造一系列匹配，初始匹配可以是空的，最后将终止在最大匹配上。一个匹配存贮在一个叫做MATE的数组中，这个数组对应每个顶点都有一个元素。如果$MATE(v)=w$，$MATE(w)=v$，那么边(v, w)就是匹配的。

当且仅当存在一条从u到v的偶数长度的交错路径时，顶点v就叫做固定非饱和顶点u的出口。很显然，这条路径$P(v)$从v到u扫描时，从一条匹配边开始。所以，$P(v)=(v,v_1,\cdots\cdots, u)$，这里$(v,v_1)$是一条匹配边。

如果扫描到一条连接外顶点v和非饱和顶点$u' \neq u$的边，那么算法找到的增广路径为

$$(u') * P(v) = (u', v, v_1, \cdots\cdots u)$$

这里$*$表示连接。如果有一条未扫描到这样的边，那么顶点u就不在增广路径中。

LABEL是个数组，它对应每个顶点有一个元素。外顶点v的LABEL元素被用来寻找交错路径P(v)。

对外顶点的LABEL元素可以是起始标号、顶点标号或者边标号。

起始标号 起始顶点u有一个起始标号，这种情况下，LABEL(u)=0。这时交错路径$P(u)=(u)$。

顶点标号 如果LABEL(v)=i，这里$1\leqslant i\leqslant n$，那么就说v有一个顶点标号。这种情况下，v是一个出口顶点，LABEL(v)是另一个出口顶点的编号。路径P(v)定义为(v, MATE(v)*P(LABEL(v))。

边标号 如果LABEL(v)=$n+2i$，$1\leqslant i\leqslant m$，那么就说v有一个边标号。此时v是一个出口顶点，LABEL(v)是连接两个出口顶点，（比如x和y）的边的编号。因此LABEL(v)=N(x,y)。顶点v的边标号$N(x,y')$表示从v到起始点u中存在一条经过边(x,y)、而具有偶数长度的交错路径$P(v)$。路径P(v)可以用P(x)和P(y)来定义。如果v在路径P(x)上，令$P(x,v)$表示P(x)上从x到v的部分。那么，$P(v)=\text{rev}P(x,v)*P(y)$，这里第一项表示从$x$到$v$路径的逆转。

当v不是一个出口顶点时，LABEL(v)<0。开始时，所有顶点都不是出口顶点，并且给这些顶点的LABEL值标为－1。

算法还用到一个叫做FIRST的数组。如果v是一个出口顶点，那么FIRST(v)就是$P(v)$中的第一个非出口顶点。如果$P(v)$不包含一个非出口顶点，那么FIRST(v)置为0。如果v是一个非出口顶点，那么FIRST(v)=0。

另一个叫做OUTER的数组是用来存贮在搜索增广路径过程中所遇到的出口顶点。搜索图就是按出口顶点在搜索中所出现的顺序而形成的。在这些出口顶点上要进行广度搜索。

Gabow算法（下面介绍）由三个程序组成，PROC-EDMONDS，PROC-LABEL，PROC-REMATCH。

PROC-EDMONDS是主程序。它是先从每个非饱和顶点开始搜索，来寻找增广路径。检测图的边，用来确定是设置标号，还是增广匹配。

当检测一条增广路径存在（算法15.5中E3）时，就调用PROC-REMATCH。这个程序计算出了一个比当前匹配多一条边的新匹配。

如果在检查边(x, y)时，出现了一个花见下面（步骤E4），那么就调用PROC-LABEL。现在x和y是外顶点（即出口顶点）。PROC-LABEL按下列操作进行：

1.变量JOIN的值被置为在$P(x)$和$P(y)$中的第一个非外顶点。

2.P(x)或P(y)中位于JOIN之前的非外顶点现在变成了外顶点，给它们设定边标号$N(x, y)$。这个边标号表示由经过边(x, y)的起始顶点到这些外顶点中的每一个顶点，都存在一条偶数长度的交错路径。

3.现在JOIN是P(x)中也是P(y)中的第一个非外点，所以，对应于P(x)和P(y)中位于JOIN前面的所有顶点的FIRST数组中的元素全部置为JOIN。

下面叙述Gabow算法，在每一步算法里，括弧中给出了适当的注释和说明。

算法15.5 最大匹配(Gabow)

FROC-EDMONDS

E0 (初始化)G 是给定的图，G的顶点编号从1到n，边编号为$n+2$，$n+4$，……，$n+2m$。设置一个虚拟顶点0。对于$0\leqslant i\leqslant n$，置LABEL(i)=−1，FIRST(i)=0。MATF(i)=0(开始时，所有顶点都是非出口和非饱和的)，则置u=0。

E1 (找非饱和顶点)置$u=u+1$。如果$u>n$，则停机；这时MATE含有一个最大匹配。否则，如果u是饱和的，则重复步骤E1。如果u是非饱和的，则把u加入OUTER数组。置LABEL(u)=0(给u设定起始标号，并开始新的一轮搜索)。

E2 (选取一条边)选取一条边(x, y)x点尚未被检查到的边(x, y)(其中x是外点)。如果不存在这样的边，则转到步骤E7。(注意：边(x, y)可以按任意顺序选取。我们采用广度搜索法：选取某个外顶点$x=x_1$，并在步骤E2的下一步循环中，选取边(x_1, y)。当所有这些边选取完毕后，在选取x_1后，立即对顶点x_2进行标号并且对于$x=x_2$重新进行上述程序。这种广度搜索需要保持一张外顶点x_1,x_2,……，的表，OUTER数组就是为此设置的)。

E3 (检测出一条增广路径的存在)如果y是非匹配的，且$y\neq u$，那么就执行PROC-REMATCH，然后转到步骤E7。

E4 (生成一个花)如果y是出口顶点，那么执行PROC-LABEL，然后转回步骤E2。

E5 (设置一个顶点标号)置v=MATE(y)。如果v是出口顶点，则转到步骤E6。如果v不是出口顶点，置LABEL(v)=x，FIRST(v)=y，并把v加入OUTER数组。在本轮搜索中，y是第一次遇到，y的对偶v是新的出口顶点。这个事实可在OUTER数组中注意到。)然后转到步骤E6。

E6 (得出下一条边)转到步骤E2。(得到一条偶长度的闭合交错路径；所以，边(x, y)不起作用。)

E7 (停止搜索)对于$0\leqslant i\leqslant n$，置LABEL(i)=−1。然后转到步骤E1。(为下一轮搜索，所有顶点都变成非外顶点)。

PROC-LABEL

L0 (初始化)置r=FIRST(x)，s=FIRST(y)。如果$r=s$，则转到步骤L6。(花中不存在非出口点)。否则，给顶点r和s做上标记。(步骤L1和L2是通过沿着路径P(x)和P(y)、用交叉前行来寻找JOIN。这种标志就是对这些路径中的非出口点r进行标号。这只需要把LABEL(r)赋给一个负的边编号即可办到，即LABEL(r)=−N(x, y)。这样，每次调用PROC-LABEL所用的标志值都不同)。

L1 (交换路径。)如果$s\neq 0$，交换r和s。(r是一个已标出的非出口顶点，交替出现在P(x)和P(y)中。)

L2 (得到非出口顶点。)置r=FIRST(LABEL(MATE(r)))。(r被置为P(x)或P(y)中的下一个非出口顶点。)如果r是未算出的，那么赋给r标志，并转到步骤L1。否则，置JOIN=r。(已求出JOIN。)转到步骤L3。

L3 （给$P(x)$、$P(y)$中顶点标号，即x和JOIN或者y和JOIN之间的所有非外顶点用边标号来标定，即$N(x, y)$。）先置v=FIRST(x)，并进行步骤L4操作；然后置v=FIRST(y)，并进行步骤L4操作。完成后转到步骤L5。

L4 （给非出口顶点v标号。）如果v=JOIN，则置LABEL(v)=$N(x, y)$和FIRST(v)=JOIN，并把v加入OUTER数组。然后置v=FIRST(LABEL(MATE(v)))。（得到下一次非出口顶点）。重复进行步骤L4操作。否则（即v=JOIN，于是对有关路径中所有非出口顶点设置边标号），按步骤L3规定的继续进行（即返回步骤L3。）

L5 （修改FIRST)对于每个出口顶点i，如果FIRST(r)是出口顶点，则置FIRST(i)=JOIN（即JOIN是$P(i)$中第一个新的出口顶点）。

L6 （边标号结束）终止程序。

PROC-REMATCH

R0 （得到增广路径）按下述步骤计算$P(x)$：

1.如果x有一个边编号$N(v, w)$，则计算$P(v)$和$P(w)$。如x位于$P(v)$上，那么

$$P(x)=(\text{rev}P(v, x))*P(w)$$

否则，

$$P(x)=(\text{rev}P(w, x))*P(v)$$

2.如果x有一个顶点标号，那么

$$P(x)=(x, \text{MATE}(x))*P(\text{LABEL}(x))$$

因此，增广路径P_a由下式给出

$$P_a=(y)*P(x)$$

R1 （增大当前匹配）从当前匹配中，移去P_a中所有匹配边，并把它加入到P_a中所有非匹配边，就得到一个新的匹配。（即，如果M是当前匹配，那么$M\oplus P_a$就是一个新的匹配。）适当修改MATE数组中元素，结束程序。 □

应当指出，在上面算法中，从某个顶点出发来寻找增广路径的搜索仅有一次。假定搜索从非饱和顶点u出发，以未找到增广路径而终止。令S_u表示这次搜索。对顶点u的Hungarian子图H是一个这样的子图：它由包含了S_u的一个外顶点的所有边和这些边上所有顶点。Edmonds在文献[15.28]中指出：在搜索S_u之后，可以不考虑Hungarian子图H。在算法15.5中可修改步骤E2如下：

E2′ （选取一条边）选取一条边，…。如果这样的边不存在，则转到步骤E1。

现在，步骤E_2′导致了步骤E7即对于S_u之后要被跳过的顶点不进行标号。

如果图没有完全匹配，这个修改将使算法加快运行。但是，它却不能改变最坏情况下的复杂性$O(n^3)$。

有关算法15.5的复杂性的讨论以及正确性的证明，可参阅文献[15.29]。

现在举例说明Gabow标法。

在图15.12(a)所示图中，对于虚线所表示的初始匹配，Gabow标法按如下进行。

起始顶点为10。在外顶点10、16、13、11和2上完成了搜索之后，搜索图如图15.12(b)所

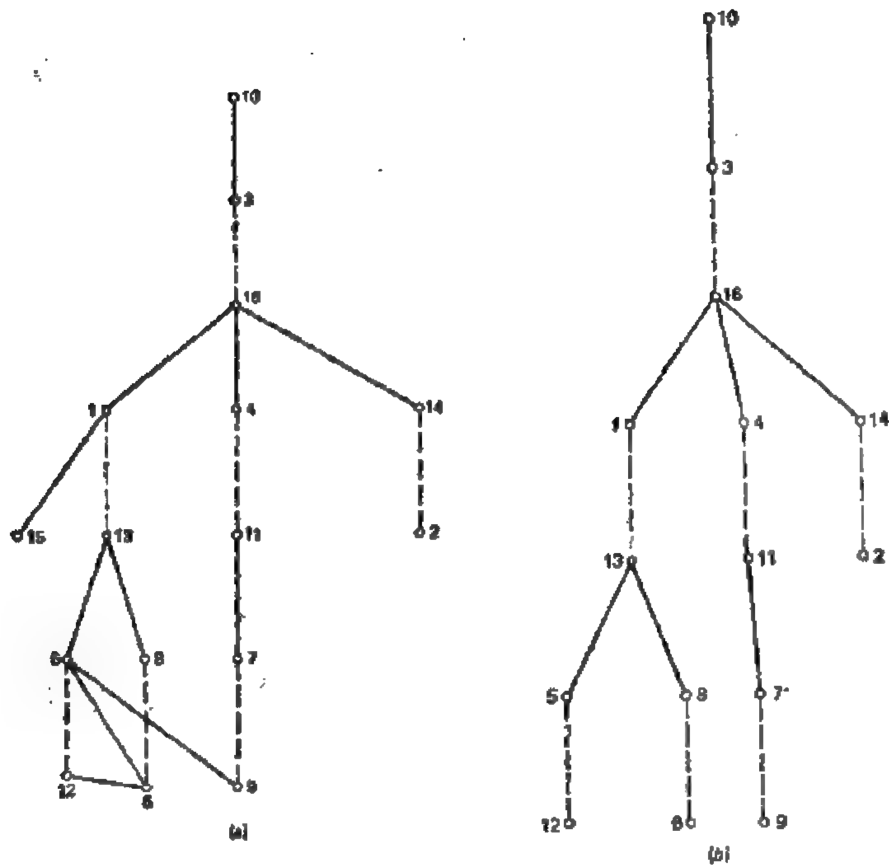


图 15.12

表 15.2

顶点	LABEL	FIRST
1	−1	0
2	16	14
3	−1	0
4	−1	0
5	−1	0
6	13	8
7	−1	0
8	−1	0
9	11	7
10	0	0
11	16	4
12	13	5
13	16	1
14	−1	0
15	−1	0
16	10	3

表 15.3

顶点	LABEL	FIRST
1	$N(9,5)$	3
2	16	14
3	−1	0
4	$N(9,5)$	3
5	$N(12,6)$	3
6	13	3
7	$N(9,5)$	3
8	$N(12,6)$	3
9	11	3
10	0	0
11	16	3
12	13	3
13	16	3
14	−1	0
15	−1	0
16	10	3

示。这一步，LABEL和FIRST数组的元素值，如表15.2所示。

OUTER数组包含的顶点顺序是10，16，13，11，2，12，6和9。

当搜索到顶点12时，检测边（12.6），从而得到花（13，5，12，6，8，13）。现在，LABEL和FIRST元素变更如下：

$$\mathrm{LABEL}(5)=\mathrm{LABEL}(8)=N(12,\ 6)$$

$$\mathrm{FIRST}(i)=1,\quad i=5,\ 12,\ 6,\ 8$$

顶点 5 和 8 加入到OUTER数组中。

其次，当搜索到顶点 6 时，检测边（6,5），从而又生成了一朵花（6，5，12，6）。但是，这朵花中的全部顶点都已是外顶点，所以没有什么可改变的了。

接着，搜索顶点 9 。检测边（9，5），从而生成了花（9，7，11，4，16，1，13,8，6，12，5，9）。这再一次改变了某些顶点的LABEL和FIRST元素值。计算的最终值，如表15.3所示。

目前,OUTER数组包含的顶点顺序是(10，16，13，11，2，12，6，9，5，8，1，7，4）。

OUTER数组中直到 9 的全部顶点均已搜索完毕，继续对剩余的顶点进行搜索。在对 5 和 8 搜索，OUTER数组中不加入任何新的顶点。当搜索到顶点 1 时，检测边（1,15），从而发现15是非饱和的。因而，找到了一条增广路径。

增广路径是$(15)*P(1)$。

用步骤R0 给出的程序来计算P（1）。顶点 1 有边编号$N(9,\ 5)$。所以计算 $P(1)$，要用到$P(9)$和$P(5)$。进而，因为顶点 5 有边编号$N(12,\ 6)$，所以计算$P(5)$要用到$P(12)$和$P(6)$。

顶点12有一个顶点标号。所以

$$\begin{aligned}P(12)&=(12,\ \mathrm{MATE}(12))*P(\mathrm{LABEL}(12))\\&=(12,\ 5)*P(13)\\&=(12,\ 5)*(13,\ \mathrm{MATE}(13))*P(\mathrm{LABEL}(13))\\&=(12,\ 5,\ 13,\ 1)*P(16)\\&=(12,\ 5,\ 13,\ 1)*(16,\ \mathrm{MATE}(16))*P(\mathrm{LABEL}(16))\\&=(12,\ 5,\ 13,\ 1,\ 16,\ 3,\ 10)\end{aligned}$$

类似地

$$P(16)=(6,\ 8,\ 13,\ 1,\ 16,\ 3,\ 10)$$

和

$$P(9)=(9,\ 7,\ 11,\ 4,\ 16,\ 3,\ 10)$$

因为顶点 5 位于$P(12)$上，

$$\begin{aligned}P(5)&=(\mathrm{rev}P(12,\ 5))*P(6)\\&=(5,\ 12)*(6,\ 8,\ 13,\ 1,\ 16,\ 3,\ 10)\\&=(5,\ 12,\ 6,\ 8,\ 13,\ 1,\ 16,\ 3,\ 10)\end{aligned}$$

现在发现顶点 1 位于P(5)上，于是

$$\begin{aligned}P(1)&=(\mathrm{rev}P(5,\ 1))*P(9)\\&=(1,\ 13,\ 8,\ 6,\ 12,\ 5)*(9,\ 7,\ 11,\ 4,\ 16,\ 3,\ 10)\\&=(1,\ 13,\ 8,\ 6,\ 12,\ 5,\ 9,\ 7,\ 11,\ 4,\ 16,\ 3,\ 10)\end{aligned}$$

从而，增广路径为

$(15) * P(1) = (15, 1, 13, 8, 6, 12, 5, 9, 7, 11, 4, 16, 3, 10)$

增广之后，我们就得到一个由边(15, 1)、(13, 8)、(6, 12)、(5, 9)、(7, 11)、(4, 16)、(3,10)和(14, 2)组成的新的M配。因为图中所有顶点在这个匹配中都是饱和的，那么它是一个最大匹配(事实上，也是一个完美匹配)。

文献[15, 33]和[15, 34]对于加权匹配问题讨论了一个有效算法。

文献[15.9]相当详细地讨论了匹配及有关问题。

15.5 二分图中的最大匹配

寻找二分图的最大匹配问题有着广泛的应用。例如，它作为求解Hitchcock运输问题文献[15.1]中的一个子问题。进而，时间表的安排在特殊情形下也包含把二分图的边集划分成互不相交的匹配。寻求划分中的元素，反过来又需要求出二分图中的最大匹配。参阅文献[15.35]。

由于它的众多应用，所以这个问题的算法复杂性便很有意义。Hopcroft及Karp在文献[15.36]中指出：如何在与$n^{5/2}$成正比的步骤内，构造出二分图的最大匹配，这里n是图中顶点的数目。

他们方法的基本思路是基于他们对匹配理论所做的、有意义的贡献。本节将讨论这些贡献和他们的二分匹配算法。

15.5.1 Hopcroft和Karp方法的基本思路

目前所给出的最大匹配算法，总是从一个匹配(不一定是最大的)开始，通过确定增广路径来得到一个具有更大基数的匹配，如果存在的话。增广路径的选取可按任何方式进行，这些算法的复杂性是$O(n^3)$。Hopcroft和Karp已证明：如果增广是沿着最短路进行，那么就可在$O(n^{5/2})$阶段内得到一个最大匹配，这里每个阶段都包括寻找一个对应于某个匹配的顶点不相交最短增广路径的极大集合。下面证明这一结论。

令M是一个匹配。如果P在所有对应于M的增广路径中的长度最小，则增广路径P叫做对应于M的最短路径。

引理15.3 设M和N是图G中两个匹配。如果$|M| = s$，$|N| = r$，且$r > s$，那么$M \oplus N$至少包含了$r - s$条对应于M顶点的不相交增广路径。

证明

考虑G在边集合$M \oplus N$上的导出了图G'。由定理8.19，G'的每一连通片或是：

1. 一个偶数长度回路，它的边交叉出现在$M-N$和$N-M$中；或是

2. 一条路径，它的边交叉出现在$M-N$和$N-M$中。

令G'的片为$C_1, C_2, \cdots, C_k$，这里每个C_i都有顶点集合V_i和边集合E_i。令

$$\delta(C_i) = |E_i \cap N| - |E_i \cap M|$$

则$\delta(C_i)$或是-1，或0，或1，对每个i，$\delta(C_i) = 1$，当且仅当C_i是对应于M的增广路径。

现在，

$$\sum_{i=1}^{k}\delta(C_i)=|N-M|-|M-N|=|N|-|M|=r-s$$

于是，G'至少有$r-s$片，使得$\delta(C_i)=1$。这些片是顶点不相交的，且每一个片都是对应于M的一条增广路径。 □

引理15.4　设M是一个匹配。令$|M|=r$，并假定最大匹配的基数是s。那么存在一条对应于M的增广路径，其长度至多为

$$2\left\lfloor\frac{r}{s-r}\right\rfloor+1$$

证明

令s是一个最大匹配，由前面的引理，$M\oplus s$至少包含$s-r$条对应于M的顶点不相交（从而也是边不相交）增广路径。这些路径所包含的M的边总共不超过r条。所以其中一条路径至多包含$\frac{r}{s-r}$条M的边，从而它至多有$2\left\lfloor\frac{r}{s-r}\right\rfloor+1$条边。 □

引理15.5　设M是一个匹配，P是对应于M的最短增广路径，而P'是对应于$M\oplus P$的增广路径。那么$|P'|\geqslant|P|+|P\cap P'|$。

证明

令$N=M\oplus P\oplus P'$。那么N是一个匹配，且$|N|=|M|+2$。因此，$M\oplus N$包含两条对应于M的顶点不相交增广路径P_1和P_2。

因为$M\oplus N=P\oplus P'$，所以$|P\oplus P'|\geqslant|P_1|+|P_2|$。但因$P$是最短增广路径，则$|P_1|\geqslant|P|$且$|P_2|\geqslant|P|$，因此$|P\oplus P'|\geqslant|P_1|+|P_2|\geqslant2|P|$。然后由等式$|P\oplus P'|=|P|+|P'|-|P\cap P'|$，则得$|P'|\geqslant|P|+|P\cap P'|$。 □

假定：从匹配$M_0=\varnothing$开始，计算一系列匹配M_1，M_2，…，M_{i+1}，…，这里$M_{i+1}=M_i\oplus P_i$，且P_i是对应于M_i的最短增广路径，那么由引理15.5，则得$|P_{i+1}|>|P_i|+|P_i\cap P_{i+1}|$。于是有下面的结论。

引理15.6　$|P_i|\leqslant|P_{i+1}|$。 □

定理15.4　对于所有的i和j，并使得$|P_i|=|P_j|$，则P_i和P_j是顶点不相交的。

证明

用引出矛盾加以证明。

假定$|P_i|=|P_j|$，$i<j$，且P_i和P_j不是顶点不相交的。那么存在k和l，并使得$i\leqslant k<l\leqslant j$，$P_k$和$P_l$不是顶点不相交的，并且对每个$r$，$k<r<l$，$P_r$是与$P_k$和$P_l$顶点不相交的。那么$P_l$是对应于$M_k\oplus P_k$的增广路径，所以$|P_l|\leqslant|P_k|+|P_k\cap P_l|$。但$|P_l|=|P_k|$，所以$|P_k\cap P_l|=0$。于是，$P_k$和$P_l$无公共边。但是，如果$P_k$和$P_l$有公共顶点$v$，那么在$M_k\oplus P_k$中与$v$相关联的边对$P_k$和$P_l$也将是公共边。于是，$P_k$和$P_l$是顶点不相交的，并且得出了一个矛盾的结果。 □

下面是本节的主要结论。

定理15.5　设s是最大匹配的基数。在序列$|P_0|$，$|P_1|$，…，$|P_i|$，…中，不同整数的数目小于或等于$2[\sqrt{s}]+2$。

证明

令 $r=\lfloor s-\sqrt{s}\rfloor$。那么 $|M_r|=r$，并由引理15.4，

$$|P_r|\leqslant 2\frac{\lfloor s-\sqrt{s}\rfloor}{(s-\lfloor s-\sqrt{s}\rfloor)}+1\leqslant 2\lfloor\sqrt{s}\rfloor+1$$

于是，对每个$i<r$，$|P_i|$是小于或等于$2\sqrt{s}+1$的$\sqrt{s}+1$个正奇整数中的一个。而$|P_{r+1}|$，…，$|P_s|$最多贡献出$s-r=\lfloor\sqrt{s}\rfloor$个不同的整数，所以序列$|P_0|$，$|P_1|$，…中不同整数的数目小于或等于$\lfloor\sqrt{s}\rfloor+1+\lceil\sqrt{s}\rceil\leqslant 2\lfloor\sqrt{s}\rfloor+2$。 □

根据引理15.6和定理15.4及15.5，可以把序列M_0，M_1，M_2…的计算看作最多由$2\lfloor\sqrt{s}\rfloor+2$个阶段组成，并使得每个阶段中找出的增广路径是顶点不相交的，且有相同长度。因为一个阶段中的所有增广路径都是顶点不相交的，所以它们也是阶段起始时匹配所对应的增广路径。这使得Hopcroft和Karp提出了下面叙述的计算最大匹配的另一方法。

S0　从某个空匹配M开始，即$M=\varnothing$。

S1　令$l(M)$是对应于M的最短增广路径的长度。找出具有下列性质的路径$(Q_1, Q_2, \cdots, Q_t)$的最大集合：

1.对于每个i，Q_i是对应于M的一条增广路径，且$|Q_i|=l(M)$。

2.Q_i是顶点不相交的。

如果不存在这样的路径，则停止。

S2　置$M=M\oplus Q_1\oplus Q_2\oplus\cdots\oplus Q_t$，转到S1。

由前面讨论可见，上述计算的S1和S2最多执行$2\lfloor\sqrt{s}\rfloor+2$次，即$O(n^{\frac{1}{2}})$次。进而，计算复杂性主要取决于实现$S_1$的复杂性。对于一般网络，实现这一步是相当复杂的，因为它需要产生对应于一个给定匹配所有的增广路径，然后再从中选出顶点不相交最短路径的最大集合。

然而，在二分图的特殊情况下，可以在$O(n^2)$时间内实现S1，并使这种特殊图的计算复杂性成为$O(n^{5/2})$。在以下部分，将讨论由Hopcroft及Karp指出的这一结果。

15.5.2 Hopcroft及Karp求最大二分匹配的方法

令G是具有二划分(X,Y)的连通二分图，我们把X中元素当作男孩，Y中元素当作女孩。

前面提到的Hopcroft及Karp的S1的实现有两步。如果匹配M不是最大的，则第一步构造出G^*，并使得对应于M的最短增广路径与G^*中起始于自由女孩、而终止于自由男孩的有向路径一一对应。第二步构造中G^*中极大的一组顶点不相交的有向路径。这些路径给出了G中对应于M所要求的最短增广路径。

图G^*构造如下：

首先给G中的边标定方向，使得对应于M的增广路径变为有向路径。这只要非匹配边从女孩指向男孩，匹配边从男孩指向女孩，即可办到。令$\overline{G}$表示形成的有向图。

令L_0为自由男孩集合，而与L_0相关联的边都是未匹配的，即从女孩指向男孩。

现在考虑女孩集合L_1，并使得每一个女孩至少与L_0中一个男孩间存在一条有向边。

如果集合L_1至少含有一个自由女孩，那么我们就已找到一条增广路径，其对应于M的一条最短增广路径长度为1。令L_1'表示L_1中自由女孩集合，E_0'表示连接L_1'和L_0的有向边。那么，要求的图G^*有一顶点集合$L_1' \cup L_0$和边集E_0'。

否则（即如果L_1不包含一个自由女孩），令E_0是连接L_1和L_2的边集合，L_2是L_1中女孩的对偶集合。令E_1是对应的匹配边。很显然L_2是男孩集合，且E_1中的边由L_2指向L_1。

通常，假定已构造出顶点集合L_0，L_1，…，L_{2k}和边集合E_0，E_1，…，E_{2k-1}。现在考虑未出现在L_0，L_1，…，L_{2k}中的女孩集合L_{2k+1}，并使得每个女孩至少与L_{2k}中一个男孩邻接。

如果L_{2k+1}至少包含了一个自由女孩，那么就找到了一条增广路径，其对应于M的一条最短增广路径长度是$2k+1$。令L_{2k+1}'是L_{2k+1}中自由女孩集合，E_{2k}'是连接L_{2k+1}'和L_{2k}的有向边集合。

否则（即L_{2k+1}不包含自由女孩），令E_{2k}是连接L_{2k+1}和L_{2k}的边集合，L_{2k+2}是L_{2k+1}中女孩的对偶集合，E_{2k+1}是相应的匹配边。

很显然，上面叙述的程序将以下列两种方式之一来结束。

1.对某个k，L_{2k+1}至少包含一个自由女孩。在这种情形下，停止程序，且要求的图G^*具有顶点集合（$L_0 \cup L_1 \cup \cdots \cup L_{2k} \cup L_{2k+1}'$）和边集合（$E_0 \cup E_1 \cup \cdots \cup E_{2k-1} \cup E_{2k}'$）。

2.集合L_{2k+1}对某个k是空的，即未得到与L_{2k}中男孩相邻接的新女孩。这表明：给定图中不存在对应于M的增广路径。于是，匹配M是最大的。

G^*的下列性质可以直接得出：

1.对于奇数i，L_i由女孩组成。否则，L_i由男孩组成。

2.对于某个i，G^*中每条边都是从L_{i+1}中某个顶点指向L_i中某个顶点。

3.G^*不含有向回路。

4.G中对应于M的最短增广路径与G^*中由某个自由女孩到某个自由男孩的有向路径一一对应。

进而，构造G^*的复杂性是$O(n^2)$。根据性质4，产生G中顶点不相交最短增广路径的最大集合等价于产生G^*自某个自由女孩到某个自由男孩的顶点不相交有向路径的最大集合，G^*中的这种有向路径可以直接在$O(n^2)$步内用深度优先搜索产生，这一点，我们可在算法15.6中看到。

现在介绍Hopcroft和Karp求最大二分匹配的算法。不失一般性，假设给定的图是连通的，算法由两个程序组成：PROC-HOPKARP和PROC-AUGMENT。

PROC-HOPKARP是主程序。给定匹配M，检查M是否为最大。如果不是，就构造图G^*并调用PROC-AUGMENT。

PROC-AUGMENT产生一个G^*中顶点不相交有向增广路径的极大集合，并进行必要的增广。在这个程序中，每一次执行时所构造的增广路径对应于在序列P_0，P_1，…的每一阶段中构造的增广路径。

在PROC-AUGMENT中，当生成一条路径时，我们就用数组PATH来存贮这个路径上的顶点。POINT是一个变量，它指出PATH中的顶部元素。

算法15.6 最大二分匹配（Hopcroft和Karp）

PROC-HOPKARP

H1 G是给定连通二分图。令M_0是零匹配，即$M=\varnothing$。置$i=0$。

H2 通过对G的每条非匹配边（对应于M_i）指定其方向为从女孩指向男孩，并且对每条匹配边（对应于M_i）指定其方向为从男孩指向女孩，来由G构造有向图$\vec{G}$。

H3 令L_0是对应于M的自由男孩集合。如果L_0非空，则转到步骤H4。否则，停止（匹配M_i是最大的）。

H4 按下面方法构造$\vec{G}$的顶点集合V的子集序列L_0，L_1，…和边集合E的子集序列E_0，E_1，…。

1. $E_i=\{(u, v) | (u, v)\in\vec{E},\ v\in L_i,\ u\notin L_0\cup L_1\cup\cdots\cup L_i\}$，$i=0, 1, 2, \cdots$。

2. $L_{i+1}=\{u |$ 对某个v，$(u, v)\in E_i\}$，$i=0, 1, 2, \cdots$。

3. $m^*=\min\{i | L_i\cap\{$自由女孩$\}\neq\varnothing\}$。

H5 如果m^*被确定，则转到步骤H6；否则，停止。（不存在对应于M_i的增广路径，故M_i是最大匹配）。

H6 构造$\vec{G}$的子图$G^*=(V^*, E^*)$，这里

$$V^*=L_0\cup L_1\cup\cdots\cup L_{m^*-1}\cup\{L_{m^*}\cup(\text{自由女孩})\}$$

以及

$$E^*=E_0\cup E_1\cup\cdots\cup E_{m^*-2}\cup\{(u, v) | v\in L_{m^*-1}\text{和}u\in\{\text{自由女孩}\}\}。$$

H7 执行程序PROC-AUGMENT，然后转到H2。

PROC-AUGMENT

A1 置POINT＝0（清除数组PATH）。选取G^*中一个尚未标上“已检”的自由女孩y，并转到步骤A2。如果不存在这样的女孩，则转到步骤A8。

A2 给y标上“已检”，置POINT＝POINT＋1和PATH(POINT)＝y（y已被放入数组PATH的顶部）。

A3 选取尚未标上“已检的边(x, y)，并转到步骤A5，如不存在这样的边（即用顶点y不能得到增广路径），则进行下列操作：

1. 若POINT＝1，则转到步骤A1。

2. 若POINT＞1，则转到步骤A4。

A4 置POINT＝POINT－2，y＝PATH(POINT)。（移去数组PATH顶部的两个元素。）转到步骤A3。

A5 给(y, x)标上“已检”，如果x标有“已检”（x已出现在某条最短增广路径，或者x不经过这种路径），则转到步骤A3。否则，给x标上“已检，并置POINT＝POINT＋1，PATH(POINT)＝x（现在x被放在数组PATH顶部）。

A6 如果x对于M_i是自由的，那么转到步骤A7。否则，令$y=x$的对偶，并转到步骤A2。

A7 现在找到了一条对应于M_i的增广路径，P的顶点是PATH(1)，PATH(2)，

…, PATH(POINT)。置$M_{i+1}=M_i \oplus P_0$（M_{i+1}是增广之后的新匹配）。置$i=i+1$，并转到步骤A1。

A8 （G^*中所有顶点不相交最短增广路径都已找到，相应的增广也已做完），终止程序。 □

可以证明，执行PROC-AUGMENT的复杂度是$O(n^2)$。因为在任何阶段，构造G^*和执行PROC-AUGMENT都仅作一次，并且根据定理15.5，至多有$O(n^{1/2})$个阶段，从而可知算法15.6的复杂性是$O(n^{5/2})$。

举例，考虑图15.13（a）所示图G，这里虚线表示匹配M中的边。图中女孩的编号从1到12，男孩的编号从－1到－10。

首先对G中的边给定方向，从而使非匹配边从女孩指向男孩，匹配边从男孩指向女孩。边的方向示于图15.13（a）。

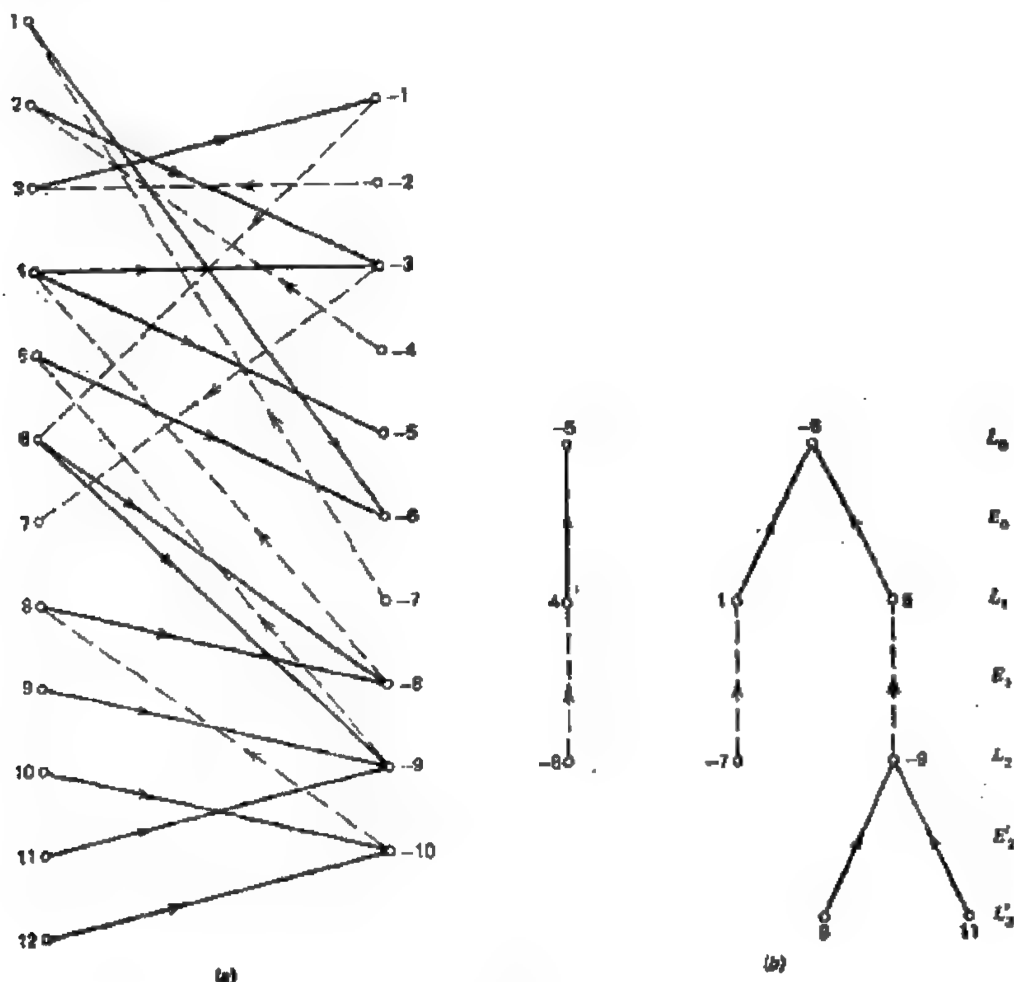


图 15.13

那么，可得集合L_0，L_1，L_2，L_3如下：

$$L_0=\{-5,\ -6\}$$

$$L_1=\{1,\ 4,\ 5\}$$

$$L_2=\{-7, -8, -9\}$$
$$L_3=\{6, 8, 9, 11\}$$

现在$m^*=3$，因为L_3的顶点6和8不是自由女孩，所以它们不会出现在图G^*中。对应于M的图G^*，如图15.13（b）所示。

注意：G^*中仅有两个增广路径。但是，它们不是顶点不相交的。所以，PROC-AUGMENT对于起始匹配M的阶段，仅产生一条增广路径。如果$\{-6, 5, -9, 9\}$是一条已产生的增广路径，那么增广之后就得到一个新的匹配，其中（$-6,5$）和（$-9,9$）是新匹配边。在M中是匹配的的边（$-5, 9$），现在变成了非匹配边。

为进一步阅读，我们推荐文献[15.37]，此文献指出：作为一般结论的特殊情形，二分图中的最大匹配可在$O(n^{5/2})$步内构造出来。

文献[15.38]对一般网络中的最大匹配，给出了一个$O(n^{5/2})$算法。这个算法用到了以广度优先搜索和深度优先搜索为基础的技术。算法还用到文献[15.29]和[15.36]早先介绍的一些思路。

15.6 完美匹配、最优分配和时间表的安排

本文将讨论最优分配和时间表的安排问题，研究这些问题包含了匹配理论。要得到一个最优分配，第一步就需要对一适当的二分图建立完美匹配。从这个角度出发，我们先讨论建立二分图的完美匹配的算法。

15.6.1 完美匹配

考虑下列人员分配问题。在n个可雇用的工人，这些工人对其中一项和n项工作是称职的。我们的兴趣在于知道我们是否能对所有这些工人分配工作，每个工人分配一项工作，并且他们对所分配的工作是称职的。在一个二分图G中，如果我们用顶点集合$X=\{x_1, x_2, \cdots, x_n\}$来表示工人，用另一个顶点集合$Y=\{y_1, y_2, \cdots, y_n\}$来表示工作。当且仅当工人$x_i$对工作$y_j$称职时，图中$x_i$与$y_j$相连。那么显然人员分配就是寻求图$G$是否有一个完美匹配。

对这个问题求解的一个方法可能是应用算法15.6来求一个最大匹配。如果这个匹配包括几条边，那么则表明图中有一个完美匹配，并且所得到的最大匹配就是完美匹配。

上述方法的主要缺点就是：如果图没有完美匹配，那么我们只能在过程结束时才知道。现在我们来讨论一种算法，它或者是求G的一个完美匹配，或者是当它发现一个满足$|\Gamma(S)|<|S|$的X的子集S时则停止，这里$\Gamma(S)$是与S中顶点相邻接的顶点集。显然，由Hall定理（定理8.13），后一种情况不存在完美匹配。

这个算法所包含的基本思路是非常简单的。通常，我们从一个初始匹配M开始。如果M使X的每个顶点都达到饱和，那么M就是我们要找的。否则，作为一般情况，我们选取X中没有饱和的顶点u，并系统地搜索一个从u出发的增广路径P。在寻找这样的路径P时，记下从集合X中所选取的顶点数、它们的邻居数，以及从集合Y中选取的顶点数。

图的二分性为我们保证了在搜索期间不会得到奇回路，从而不会开花。正如我们在

15.4.1节中所见，一个偶数长度的闭合交错路径对增广给定的M匹配是没有用的。因此，我们正在展开的搜索图总是一个树。这棵树叫做Hungarian树。在任何阶段，如果发现了一条增广路径，就完成了其增长并且得到一个新的匹配，这个新匹配使X中的饱和顶点又多了一个，从而步骤如前进行。如果这样的路径在存在，那么就能得到一个集合$S\subseteq X$，从而破坏了完美匹配存在的充要条件。

令M为G中的一个匹配，如果u为X中的一个未饱和顶点。

1.u属于H的顶点集合；并且

2.对于H的每个顶点v，从H中u到v的唯一路径是M-交错路径（即一条与M相交的交错路径）。那么，G中的树H称为以u为根的M-交错树。

我们用S表示X的顶点子集，并用T表示H中出现的Y的顶点子集。

交错树按如下过程生成。起初，H只包含顶点u，然后就以下列方式增长：在任何阶段里都存在两种可能性：

1.H中除u以外的所有顶点被饱和（例如，见图15.14（a））。

2.H包含一个异于u的未饱和顶点（例如，见图15.14（b））。在这种情况下，存在一条增广路径，从而可得一个新匹配。

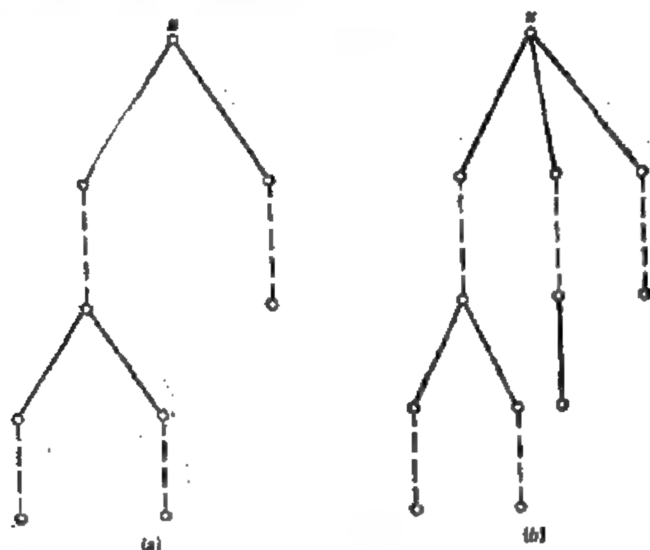


图 15.14 交错树的例子

在第一种情况下，或者$\Gamma(S)=T$，或者$T\subset\Gamma(S)$。

1a.$\Gamma(S)=T$。因为在树H中，$|S|=|T|+1$，所以在这种情况，我们有$|\Gamma(S)|=|S|-1$，因而集合S不满足Hall定理所要求的充要条件。因此得出结论：G中没有完美匹配。

1b.$T\subset\Gamma(S)$。所以在Y中存在一个顶点，但不出现在T中，却出现在$\Gamma(S)$中，令该顶点Y与S中的顶点x相邻接。如果y被饱和，并以顶点Z作为其配偶，那么就通过增加顶点y和z，以及边（x，y）和（y，z）来增广H。这时，我们就回到了第一种情况。如果y是未饱和的，则通过增加顶点y和边（x，y）来增广H，就得到了第二种情况。H中从

u到y的路径是一条与M相关的增广路径。

以上所描述的方法表示在下列算法中。

算法15.7 完美匹配

S1 令G为具有二划分(X, Y)、且$|X|=|Y|$的二分图，令M_0为空匹配，即$M_0=\varnothing$。置$i=0$。

S2 如果在匹配M_i中，X中的所有顶点都被饱和，则停止。(M_i是G的完美匹配。)否则，选取X中的一个未饱和顶点u，并置$S=\{u\}$，$T=\varnothing$。

S3 如果$\Gamma(S)=T$，则停止。(现在$|\Gamma(S)|<|S|$，从而G中无完美匹配。)否则，从$\Gamma(S)-T$中选取一个顶点y。

S4 如果y在M_i中未被饱和，则转到S5。否则，置$z=y$的配偶，$S=S\cup\{z\}$，$T=T\cup\{y\}$，然后转到S3。

S5 (得到了一条增广路径)置$M_{i+1}=M_i\oplus P$，$i=i+1$。转到S2。 □

举例，考虑图15.15(a)所示的二分图G。图中，初始匹配M的边用虚线表示。顶点x_1在M中未饱和。现在来展开以x_1为根的M交错树。当我们定下了增广通路x_1、y_1、x_2、y_2的位置，我们就终止这棵树的增广，见图15.15(b)。然后，增广M就得到了新的匹配，见图15.15(c)。

在这一新匹配中，x_4未被饱和。因此，对此新的匹配，我们着手展开以x_4为根的交错树。这棵交错树如图15.15(d)所示而终止。因为在这一阶段，$\Gamma(S)=T$，这里$S=\{x_1, x_2, x_3, x_4\}$，$T=\{y_1, y_2, y_3\}$，所以这棵交错树不可能进一步增广。因此图15.15(a)所示图没有完美匹配。

有时，我们对于求具有特殊性质的完美匹配感兴趣。文献[15.39]讨论了用于此类问题的一个算法。

15.6.2 最优分配

考虑这样一种分配问题，其中每个工人对各项工作都称职。这里显然可以给每个工人分配一项工作，(当然，我们如前假设，有n个工人和n项工作)。事实上，任何一个最大匹配都可以做到这一点，并且可得$n!$个这样的匹配。在这种情况下，我们所感兴趣的问题是，要考虑工人对其工作的效率问题，然后进行分配，以使工人的总效率达到最大。求这样的一种分配问题，称为最优分配问题。

这个问题的二分图是一个完全二分图，即如果$X=\{x_1, x_2, \cdots, x_n\}$代表工人，$Y=\{y_1, y_2, \cdots, y_n\}$代表工作，那么对于所有的$i$和$j$，$x_i$与$y_j$相邻接。同时，我们对每条边($x_i$, y_j)赋予一个权$w_{ij}=w(x_i, y_j)$，用来代表工人x_i对工作y_j的效率(以某个单位量度)。那么，在此加权图中，最优分配问题对应于求一个最大权完美匹配，这样的一个匹配称为最优匹配。

现在讨论一个求最优匹配问题的$O(n^4)$算法。该算法是由Kuhn和Munkrer在文献[15.40]、[15.41]中提出的。我们根据文献[15.35]所给出的论述来进行讨论。

一个可行的顶点标号是一个在集合$X\cup Y$上的实值函数f，并使得

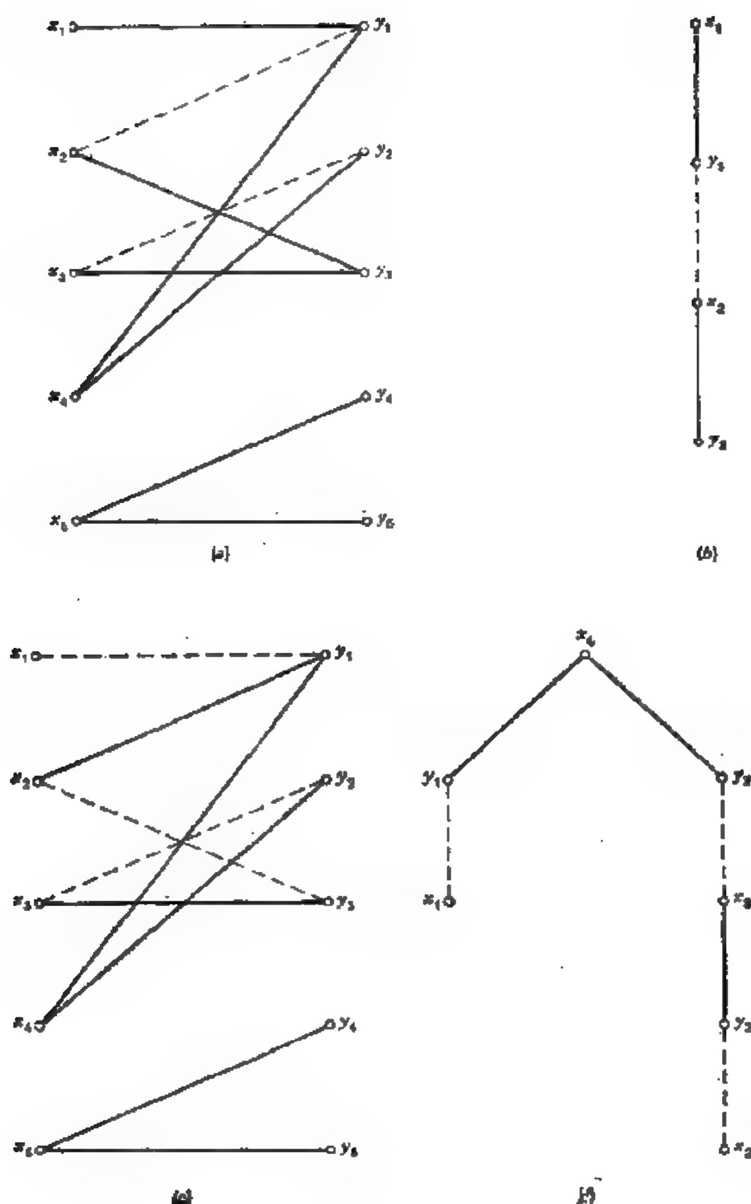


图 15.15

$$f(x)+f(y)\geqslant w(x,y),\ \text{对所有}x\in X\text{和}y\in Y$$

那么，$f(x)$ 就叫做顶点 x 的标号。

例如，下面的标号是一个可行的顶点标号函数：

$$f(x)=\max\{w(x,y)\},\quad \text{若}x\in X$$

$$f(y)=0\qquad\qquad\qquad \text{若}y\in Y$$

显而易见，这里总是存在一个无论权值大小的可行的顶点标号。

对于某一给定的可行顶点标号 f，令 E_f 表示G中所有满足 $f(x)+f(y)=w(x,y)$ 的那些边 (x,y) 的集合。具有边集合 E_f 的 G 的生成子图叫做与 f 对应的等价子图。我们用 G_f 表示这个子图。

下面讨论等价子图和最优匹配所关联的定理构成了Kuhn-Munkres算法的基础。

定理15.6 令f为图$G=(V,E)$的一个可行顶点标号函数。如果G_f包含一个完美匹配M^*，那么M^*就是G的一个最优分配。

证明

假定G_f包含一个完美匹配M^*。因为G_f是G的一个生成子图，所以M^*也是G的一个完全匹配。令$w(M^*)$表示M^*的权，即

$$w(M^*)=\sum_{e\in M^*}w(e)$$

因为每条边$e\in M^*$都属于等价子图，且M^*中边的顶点恰好覆盖G的每个顶点一次，所以得到

$$w(M^*)=\sum_{e\in M^*}w(e)=\sum_{v\in V}f(v) \tag{15.28}$$

另一方面，如果M是G中任一完美匹配，那么

$$w(M)=\sum_{e\in M}w(e)\leqslant\sum_{v\in V}f(v) \tag{15.29}$$

综合式(15.28)和(15.29)，即得

$$w(M^*)\geqslant w(M)$$

因此，M^*是G的一个最优匹配。 □

在Kuhn-Munkres算法中，我们首先从任意一个可行的顶点标号函数f开始，来求相应的G_f。选取G_f中的一个初始匹配M来应用算法15.7。如果在G_f中可得一个完美匹配，那么由定理15.6，该匹配就是最优的。否则，算法15.7就终止于一个非完美的匹配M'，同时给出一个M'-交错树H，它不包含M'-增广路径，且在G_f中无法进一步增广。然后我们利用M'和H包含于G_f中这一性质来修改f，得到一可行的顶点标号f'，并且在$G_{f'}$中增广H。只要需要，我们就对可行的顶点标号做这种修改，直到在某个等价子图中找到一个完美匹配为止。Kuhn-Munkres算法的详细叙述如下：

算法15.8 最优分配(Kuhn-Munkres)

S1 G为给定的、具有二划分(X,Y)、且$|X|=|Y|$的完全二分图，$W=[w_{ij}]$为给定的权矩阵。置$i=0$。

S2 从G中任一可行顶点标号f开始，求等价子图G_f，然后选取G_f中的一个初始匹配M_i。

S3 如果X中的所有顶点在M_i中都被饱和，那么M_i就是一完美匹配，从而由定理15.6可知，它是一最优匹配，就此停止。否则，令u为X中的一个非饱和顶点，置$S=\{u\}$，$T=\varnothing$。

S4 令$\Gamma_f(S)$为G_f中与S中的顶点相邻接的顶点集合。如果$\Gamma_f(S)\supset T$，那么转到$S5$。否则，(即，若$\Gamma_f(S)=T$)计算

$$d_f=\min_{\substack{x\in S\\ y\notin T}}\{f(x)+f(y)-w(x,y)\} \tag{15.30}$$

并且得到一个新的可行顶点标号f'：

$$f'(v)=\begin{cases}f(v)-d_f, & 若v\in S\\ f(v)+d_f, & 若v\in T\\ f(v), & 其它\end{cases} \tag{15.31}$$

（注意，$d_f>0$且$\Gamma_f(S)=T$。）

用f'代替f，$G_{f'}$代替G_f。

S5 从$\Gamma_f(S)-T$中先取顶点y。如果y在M_i中未被饱和，则转到S2。否则，置$z=M_i$中y的对偶，$S=S\cup\{z\}$且$T=T\cup\{y\}$，然后转到S4。

S6 （得到了一个增广路径P。）置$M_{i+1}=M_i\oplus P$和$i=i+1$，转到S3。 □

为举例说明Kuhn-Munkres算法，考虑一个具有下列权矩阵$W=[w_{ij}]$的完全二分图G：

$$W=\begin{bmatrix} 4 & 4 & 1 & 3 \\ 3 & 2 & 2 & 1 \\ 5 & 4 & 4 & 3 \\ 1 & 1 & 2 & 2 \end{bmatrix}$$

G的一个初始可行顶点标号f可选取如下：

$$f(x_1)=4, f(x_2)=3, f(x_3)=5, f(x_4)=2$$

$$f(y_1)=f(y_2)=f(y_3)=f(y_4)=0$$

等价子图G_f，如图15.16(a)所示。应用算法15.6，我们发现G_f没有完美匹配，这是因为对于集合$S=\{x_1, x_2, x_3\}$，$T=\Gamma(S)=\{y_1, y_2\}$。

利用式（15.30），可计算出

$$d_f=1$$

然后用式（15.31）就得到了如下新的标号函数f'：

$$f'(x_1)=3,\ f'(x_2)=2,\ f'(x_3)=4,\ f'(x_4)=2$$

$$f'(y_1)=1,\ f'(y_2)=1,\ f'(y_3)=0,\ f'(y_4)=0$$

等价子图$G_{f'}$，如图15.16(b)所示。对于$G_{f'}$应用算法15.6，则得到由边(x_1, y_2)、(x_2, y_1)、(x_3, y_3)、(x_4, y_4)组成的完美匹配M。这个匹配就是最优匹配。

Megido和Tamir在最近的文献[15.42]中，针对一类带权匹配在排进度和最优分配的某些应用中产生的问题，讨论了一个$O(n\log n)$算法。

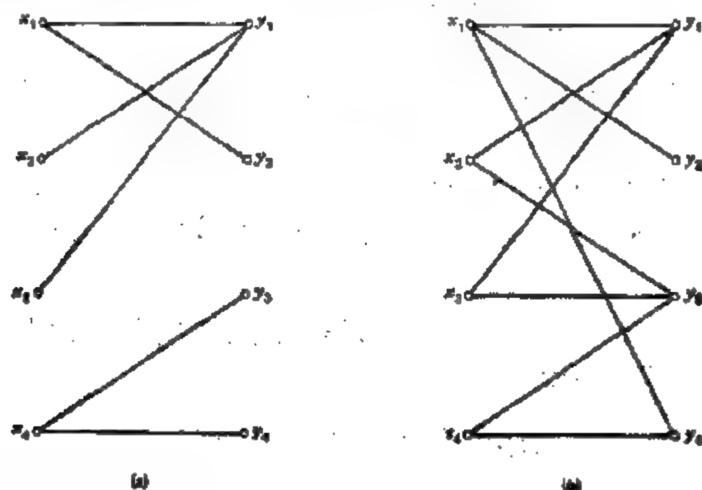


图 15.16

15.6.3 时间表安排

在一所学校中有p个教师x_1, x_2, …, x_p和q个班级y_1, y_2, …, y_q。要求教师x_i教班

级y_j，学时为p_{ij}。我们希望安排一张时间表，它具有最小可能的学时数。这就是被称为时间表安排问题的一个特例。

假定我们建立了二分图$G=(X, Y)$，其中X中的顶点代表教师，Y中的顶点代表班级，以及顶点$x_i \in X$通过p_{ij}条平行边与$y_j \in Y$连接。因为在一个课时内，一个教师至多只能教一个班级，一个班级至多只能由一位老师来教。这样就得出了一个学时的课程表与G中一个匹配相对应。反之，在一个课时内，每个匹配对应于教师对班级的可能安排。因此，上面提到的时间表安排问题，就是把G的边划分为尽可能少的匹配。

由推论8.22.1可知，在一二分图G的边集的任一划分中，最小匹配数等于G的最大度数。该定理的证明也给出了下面确定具有最小匹配数的划分过程：

S1 令G为给定的二分图。置$i=0$，$G_0=G$。

S2 构造G_i的一个匹配M_i，并使得G_i中所有最大度数顶点被饱和。

S3 从G_i中删去M_i，令G_{i+1}表示其结果图。如果G_{i+1}没有边，那么M_0，M_1，…，M_i就是一个所求的图G边集合的划分。否则，置$i=i+1$，并转到S1。

很显然，上述过程的复杂性取决于实现匹配M_i的S2的复杂性，这个匹配M_i使二分图$G=(X,Y)$中所有的最大度数顶点均被饱和。这样的一个匹配可按下法求得。（见定理8.21的证明。）

令X_a表示X中最大度数顶点的集合，Y_b为Y中最大度数顶点的集合。令G_a表示由X_a中顶点的关联边所构成的G的子图。类似地，G_b为由Y_b中顶点的关联边所构成的G的子图。

由定理8.22可知，在X_a中存在一个匹配M_a，使X_a中所有顶点饱和，匹配M_a是G_a中的一个最大匹配。类似地，在Y_b中存在一最大匹配M_b，它使Y_b中所有顶点饱和。根据使用定理8.21的过程，我们可以从M_a和M_b中求得一匹配M，并使得X_a和Y_b中的顶点饱和。M即为所求的、使G中所有最大度数顶点均饱和的匹配。

求M_a和M_b的复杂性与求一个最大匹配的复杂性相同，即$O(n^{5/2})$，这里n为G的顶点数目。很容易证明，由M_a和M_b构造M的复杂性为$O(n^2)$。因此，实现S2的总复杂性为$O(n^{5/2})$。

因为S2将要重复Δ次，这里Δ为G中最大度数，且$\Delta \leqslant n$，所以构造所要求的时间表的复杂性为$O(n^{7/2})$。

关于时间表安排问题一般形式的讨论及其与此有关的参考书，可在文献[15.43]中找到。

15.7 运输网络中的流

一个运输网络N是满足以下条件的无自环连通有向图：

1.仅有一个具有零入度的顶点，称之为源，并用s表示。

2.仅有一个具有零出度的顶点，称之为汇，并用t表示。

3.N中每一条有向边$e=(i,j)$都被赋予一非负实数，称为该边的容量，用$c(e)$或$c(i,j)$表示。如果不存在从i指向j的边e，那么定义$c(e)=0$。

运输网络N描述了一种商品沿着交通路线从产地中心到市场的运输模型，因而可把边容量认为是商品沿该边所能运输的最大速率。

运输网络N中的流f是分配给每条边$e=(i,j)$一个非负实数$f(e)=f(i,j)$的分配，并满足以下条件：

$$1.\ f(i,j)\leqslant c(i,j)，对于N中每条边(i,j) \tag{15.32}$$

$$2.\ \sum_{所有i} f(i,j)=\sum_{所有j} f(j,i)，对于所有i\neq s,t \tag{15.33}$$

边e的流值$f(e)$可以认为是在流f的情况下、物资沿e运输的速率。条件(15.32)称为容量约束，它要求边流的速率不能超过该边的容量。条件式(15.33)称为守恒条件，它要求对于除源与汇以外的每个顶点i，物资输入i的速率必须等于物资从i输出的速率。

举例，一个具有流f的运输网络，如图15.17所示。图中，在每条边e的旁边顺序标出了容量$c(e)$和流量$f(e)$。

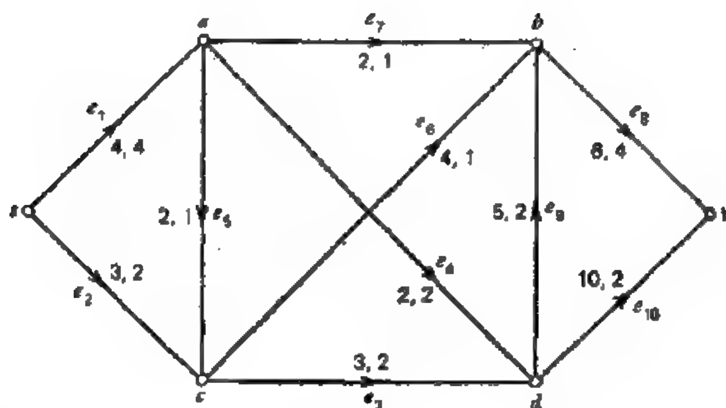


图 15.17 运输网络

流f的值用val(f)表示，其定义为

$$\mathrm{val}(f)=\sum_{所有j} f(s,j) \tag{15.34}$$

运用守恒条件，我们很快将证明：

$$\mathrm{val}(f)=\sum_{全部j} f(s,j)=\sum_{全部j} f(j,t) \tag{15.35}$$

上式仅仅说明了一个直观而又明显的事实，即在守恒条件下，从源点输出的物资总量等于输入汇点的物资总量。

如果在N中不存在f，并使得val$(f)>$val(f^*)。那么运输网络中的流f^*就认为是最大的。

在下一小节，我们将建立Ford-Fulkerson的最大流最小切割定理，然后讨论求运输网络中最大流的Ford-Fulkerson标号方法。最后用最大流最小切割定理证明Menger定理。

15.7.1 最大流最小切割定理

运输网络N的一个切割$\langle S,\bar{S}\rangle$，如果$s\in S$且$t\in\bar{S}$，则称之为是分离开源 s 与汇 t 的，这样的切割记为$s-t$切割。切割$K=\langle S,\bar{S}\rangle$的容量$c(K)=c(S,\bar{S})$定义为

$$c(K)=\sum_{\substack{i\in S\\ j\in\bar{S}}}c(i,j)$$

注意：从$\bar{S}$指向S的边的容量对于切割$K=\langle S,\bar{S}\rangle$的容量没有贡献。我们用$f(S,\bar{S})$表示从$S$指向$\bar{S}$的所有边的流量和，量值$f(\bar{S},S)$也以类似的方式定义。

例如，参看图15.17所示的运输网络的切割$K=\langle S,\bar{S}\rangle$，其中

$$S=\{a,b,c,s\},\quad \bar{S}=\{d,t\}$$

从S指向$\bar{S}$的边是e_3、e_4、e_6。因此

$$c(S,\bar{S})=c(e_3)+c(e_4)+c(e_6)=11$$

$$f(S,\bar{S})=f(e_3)+f(e_4)+f(e_6)=8$$

$$f(\bar{S},S)=f(e_5)=2$$

定理15.7 对于运输网络N中任一流f和任一$s-t$切割$\langle S,\bar{S}\rangle$，

$$\mathrm{val}(f)=f(S,\bar{S})-f(\bar{S},S)\tag{15.37}$$

证明

根据流和流值的定义，我们有

$$\sum_{所有j}f(i,j)-\sum_{所有j}f(j,i)=\begin{cases}\mathrm{val}(f), & i=s\\ 0\ , & i\in S-\{s\}\end{cases}$$

遍取S中全部顶点代入上式，并将所得方程对应相加得：

$$\sum_{i\in S}\ \sum_{所有j}f(i,j)-\sum_{i\in S}\ \sum_{所有j}f(j,i)=\mathrm{val}(f)\tag{15.38}$$

在上面的方程中，当$i\in S$，$j\in S$时，$f(i,j)$与$-f(i,j)$各自都在左端正好出现一次，即并被抵消了。因此，式(15.38)化简为

$$\sum_{i\in S}\sum_{j\in\bar{S}}(i,j)-\sum_{i\in S}\sum_{j\in\bar{S}}f(j,i)=\mathrm{val}(f)\tag{15.38}$$

即
$$f(S,\bar{S})-f(\bar{S},S)=\mathrm{val}(f)$$

注意，式(15.34)是式(15.37)的特例。

推论 15.7.1 对于运输网络N中任一流和任一$s-t$切割$K=\langle S,\bar{S}\rangle$，

$$\mathrm{val}(f)\leqslant c(S,\bar{S})\tag{15.39}$$

证明

因为每个$f(i,j)$都是非负的，所以由式(15.37)可得

$$\mathrm{val}(f)=f(S,\bar{S})-f(\bar{S},S)\leqslant f(S,\bar{S})\leqslant c(S,\bar{S})$$ □

如果$f(i,j)=c(i,j)$，则边(i,j)称为f-饱和的；否则称为f-非饱和的。如果$f(i,j)>0$，则边(i,j)称为f-正值的；如果$f(i,j)=0$，则边(i,j)称为f-零值的。

注意：式(15.39)的等号成立，当且仅当$f(\bar{S},S)=0$和$f(S,\bar{S})=c(S,\bar{S})$。换言之，$\mathrm{val}(f)=c(S,\bar{S})$，当且仅当从$S$指向$\bar{S}$的所有边都是$f$-饱和的，以及从$\bar{S}$指向$S$的所

有边都是f-零值的。

对于运输网络N中的一个$s-t$切割K，如果N中不存在$s-t$切割K'，并使得$c(K')<c(K)$，那么$s-t$切割K就是最小切割。

推论15.7.2 设f为一流，K为一$s-t$割，从而$\mathrm{val}(f)=c(K)$。那么f就是最大流，K就是最小$s-t$切割。

证明

设f^*为最大流，K^*为最小$s-t$切割。根据推论15.7.1，有

$$\mathrm{val}(f^*)\leqslant c(K^*)$$

所以，我们有

$$\mathrm{val}(f)\leqslant \mathrm{val}(f^*)\leqslant c(K^*)\leqslant c(K)$$

因为由假设已知$\mathrm{val}(f)=c(K)$，所以$\mathrm{val}(f)=\mathrm{val}(f^*)=c(K^*)=c(K)$。从而得$f$是最大流，$K$是最小$s-t$切割。 □

下面证明，最大流的值实际上等于最小切割的容量。

考虑一个具有流f的运输网络N。令



是N中从源到某顶点v的一条路径。注意：P不一定是一条有向路径。

如果P的边e_i从u_{i-1}指向u_i，那么P的边e_i称为P的前向边反之称为P的反向边。对于P的一条边e_i，令

$$\varepsilon_i(P)=\begin{cases}c(e_i)-f(e_i), & \text{若}e_i\text{是前向边}\\ f(e_i), & \text{若}e_i\text{是反向边}\end{cases}\qquad (15.40)$$

同时将非负实数$\varepsilon(P)$赋予路径P，而$\varepsilon(P)$定义为

$$\varepsilon(P)=\min_i\{\varepsilon_i(P)\}\qquad (15.41)$$

注意：$\varepsilon(P)\geqslant 0$。

如果一条路径的所有前向边是f-非饱和的，且所有反向边是f-正值的，那么该路径就称为f-非饱和的。

如果一条$s-t$路径P是f-非饱和的，那么P称为f-可增路径。由式（15.40）和（15.41）可得，对于一条$s-t$路径P，$\varepsilon(P)>0$；当且仅当P是f-可增的。在给定网络N的一条$s-t$路径P后，我们可定义一个新的流$\hat{f}$：

$$\hat{f}(e)=\begin{cases}f(e)+\varepsilon(P), & \text{若}e\text{是}P\text{的前向边}\\ f(e)-\varepsilon(P), & \text{若}e\text{是}P\text{的反向边}\\ f(e), & \text{其它}\end{cases}$$

显而易见，

$$\mathrm{val}(\hat{f})=\mathrm{val}(f)+\varepsilon(P)$$

因此，$\mathrm{val}(\hat{f})>\mathrm{val}(f)$，当且仅当$P$是$f$-可增的。换言之，如果存在一条$f$-可增路径，那么流$f$就不是最大的。

举例，考虑图15.17中的网络N。设流f如图所示。在由e_2、e_5、e_7和e_8组成的路径P上，边e_2、e_7、e_8是前向边，e_5是唯一的后向边。对于流f，$\varepsilon_2(P)=1$，$\varepsilon_5(P)=1$，$\varepsilon_7(P)=1$，$\varepsilon_8(P)=2$。因此，$\varepsilon(P)=\min\{\varepsilon_2(P),\ \varepsilon_5(P),\ \varepsilon_7(P),\ \varepsilon_8(P)\}=1$。

因为$\varepsilon(P)>0$，所以P是f-可增的，图15.18给出了路径P上的修正流$\hat{f}$。应当指出：$\hat{f}$是通过在P的所有前向边中增加$\varepsilon(P)$，以及在P的所有后向边中减去$\varepsilon(P)$而得到，其余边流均保持不变。

定理15.8 运输网络N的一个流f是最大的，当且仅当不存在f-可增路径。

证明

必要性 如果N中存在一条f-可增路径P，那么f显然不是最大流，这是因为P上的修正流$\hat{f}$具有比f更大的值。

充分性 假定N中不存在f-可增路径。令S表示从源开始沿f-可增路径所能到达的全部顶点的集合。很显然，$s\in S$。进而因为N中不含f-可增路径，所以$t\in\bar{S}$。

现在来说明 $\mathrm{val}(f)=c(S,\ \bar{S})$，并由此证明(根据推论15.7.2)$f$是最大流以及$(S,\ \bar{S})$是最小切割

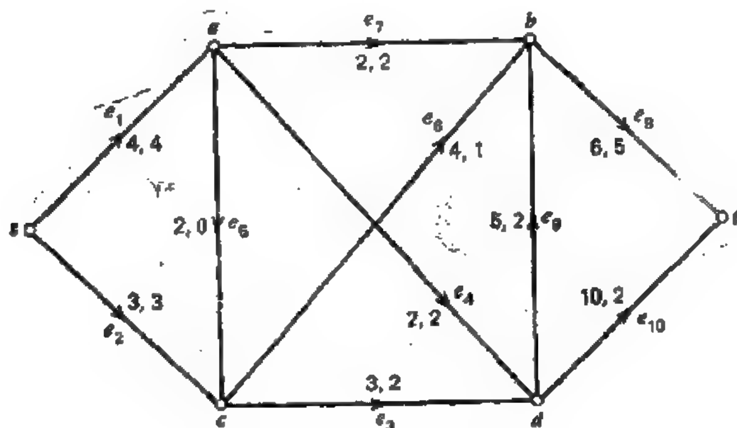


图 15.18

考虑一有向边(v,w)路，其中$v\in S$，$w\in\bar{S}$。由于$v\in S$，则存在一条f-非饱和的$S-v$径Q，且边$(v,\ w)$必为f-饱和的。不然的话，可将Q延伸来形成一条f-非饱和的$s\in w$路径，但$w\in\bar{S}$，故这是不可能的。类似地，我们可以证明每条从$\bar{S}$指向S的边$(v,\ w)$都应是f-零值的。因此，$f(S,\ \bar{S})=c(S,\bar{S})$以及$f(\bar{S},\ S)=0$，即有

$$\mathrm{val}(f)=f(S,\ \bar{S})-f(S,\ \bar{S})=c(S,\ \bar{S})。$$

再由推论15.7.2可知，f是最大流，$\langle S,\ \bar{S}\rangle$是最小切割。 □

在以上定理的证明过程中，我们业已推出了下面由Ford-Fulkerson在文献[15.44]中和文献[15.45]所给出的重要结果。

定理15.9 （**最大流最小切割定理**） 在一运输网络中，最大流的值等于最小切割的容量。 □

最大流最小切割定理可以用来证明大量具有组合意义的结果。在15.7.4中，我们将讨论这些结果之一，其余请参阅文献[15.1]和[15.46]。

15.7.2 Ford-Fulkerson标号算法

本小节讨论由Ford和Fulkerson提出的、求运输网络最大流的算法，这个算法（根据定理15.8）由两个阶段组成。

给定流 f，在第一阶段，我们用一个标号过程来检查是否存在一条f-可增路径。如果这样的路径不存在，那么由定理15.8可知，给定流f就是最大的。否则，转入第二阶段。在第二阶段，我们利用在第一阶段中产生的标号，求出一条f-可增路径P，同时得出P上的修正流$\hat{f}$。然后，再对新流$\hat{f}$重复第一阶段。注意 $\mathrm{val}(\hat{f})>\mathrm{val}(f)$。

在第一阶段，将(d_v, Δ_v)的标号给予顶点v。标号中的第一个符号d_v指出了v的前一个顶点，即v从该顶点获得自己的标号，d_v也说明了标号的方向——前向或后向。读者很快就会看到，标记了顶点v，就意味着存在一条f-非饱和的$s-v$路径P，并且对于此路径P，$\varepsilon(P)=\Delta_v$。

第一阶段是以把源s标为$(-, \infty)$来开始的，这里d_s的值是任意的。以后顶点的标号按以下原则进行：

设顶点u已被标号，而顶点v尚未标号。令e是连接u和v的边

前向标号：如果$e=(u,v)$，那么只要$c(e)>f(e)$，从u开始沿e对v进行前向标号就是可行的。若此标记完结后，v就得到标号(u^+, Δ_v)，这里

$$\Delta_v=\min\{\Delta_u, c(e)-f(e)\}$$

后向标号：如果$e=(v, u)$，那么只要$f(e)>0$从u开始沿e对v进行后向标号就是可行的。若此标号完结后，v就得到标号(u^-, Δ_v)，这里

$$\Delta_v=\min\{(\Delta_u, f(e)\}$$

在第一阶段，一个顶点至多被标号一次。在当（1）顶点t得到标号后，或者（2）t未被标号，即不再存在可能被标记的顶点时，第一阶段就此结束。

如果 t 在第一阶段得到了标号，那么由标号规则可知，存在一条f-可增路径P，并且$\varepsilon(P)=\Delta_t$。在第二阶段，利用符号d_v，沿路径P反溯回去，从而求得P上的修正流$\hat{f}$。如果第一阶段在 t 未被标记的情况下结束，那么就意味着不存在f-可增路径，因此，给定流 f 就是最大的。

下面叙述Ford-Fulkerson算法。

算法15.9 运输网络中的最大流（Ford-Fulkerson）

S1 在给定运输网络N中选择任一流 f。对于N中各边e，选定$f(e)=0$。

S2 （第一阶段开始）将s标记为$(-, \infty)$。

S3 若有（可以通过前向标记或后向标记来定）未标记顶点时，那么就选出这样一个顶点v进行标号，然后转S至4。否则，转到S7。

S4 如果$v=t$，那么转到S5（第一阶段结束）。否则，返回S3。

S5 （第二阶段开始）设v的标号是(d_v, Δ_v)。现在做以下工作：

1. 如果$d_v=u^+$，那么置$f(u,v)=f(u,v)+\Delta_t$，
2. 如果$d_v=u^-$，那么置$f(v,u)=f(v,u)-\Delta_t$。

S6 如果$u=s$，则删去所有标号（第二阶段结束），并转到S2。否则，置$v=u$，并转到S5。

S7 （选定流f即为最大的）停机。 □

下面我们举例说明上述算法。

考虑图15.19所示的运输网络N。图中，在每条边e的旁边顺序标出了$c(e)$和$f(e)$。对于N的所有边，取初始流$f(e)=0$。开始把源s标为（$-$，∞），然后顺序标记（算法15.9的S3）顶点a，b，c，d和t，所得各标号为

$$a:\ (s^+,\ 3)$$
$$b:\ (a^+,\ 3)$$
$$c:\ (a^+,\ 3)$$
$$d:\ (c^+,\ 2)$$
$$t:\ (b^+,\ 2)$$

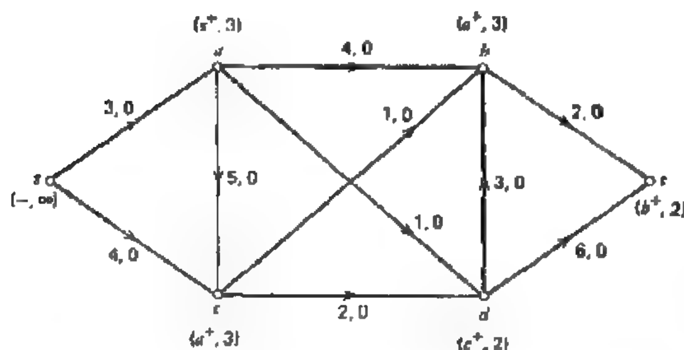
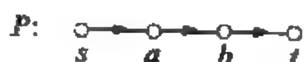


图 15.19 Ford-Fulkerson标号算法举例用网络

因为顶点t正好被标记了，所以第一阶段结束。在第二阶段，我们按以下方法（S5、S6）求得可增路径P和修正流f_1。

t的最初标号是b^+，这说明：在路径P上，顶点b在t的前面。b的最初标号表明：在路P上，a在b的前面。类似地，可以看到s在a之前。因此



P中的所有边都是前向边。所以为了求得修正流f_1，我们把P的每条边的流值增加$\Delta_t=2$，这样f_1的值则等于2，如图15.20(a)所示。

现在删去全部顶点的标号。如从f_1开始，则得一个如图15.20(a)所示的新的标号集。对应f_1的一条可增路径由顶点s、c、d、t顺序组成，这条路径P的所有边均为前向边，这些边的流值又增加了2。所得的流f_2，如图15.20(b)所示。

图15.20(b)还给出了基于f_2的新标号集。对应于f_2的一条可增路径由s、c、b、a、t组成。连接a和b的边是路径上的一条后向边，其余的边均为前向边。现在将前向边的流增加1，而后向边的流减少1。这样得到的流f_3如图15.20(c)所示。

再从f_3入手继续标记顶点。这一过程在图15.20(c)中不存在标记t的情况下结束。因

此，不存在对应于流f_3的可增路径。因而，f_3是最大流。

设S为图15.20(c)中已标记过顶点的集合。因此，$S=\{s, a, b, c\}$。根据定理15.8的证明可知，切割$\langle S, \bar{S}\rangle$为一最小切割，且$\text{val}(f_3)=\langle S, \bar{S}\rangle$的容量。

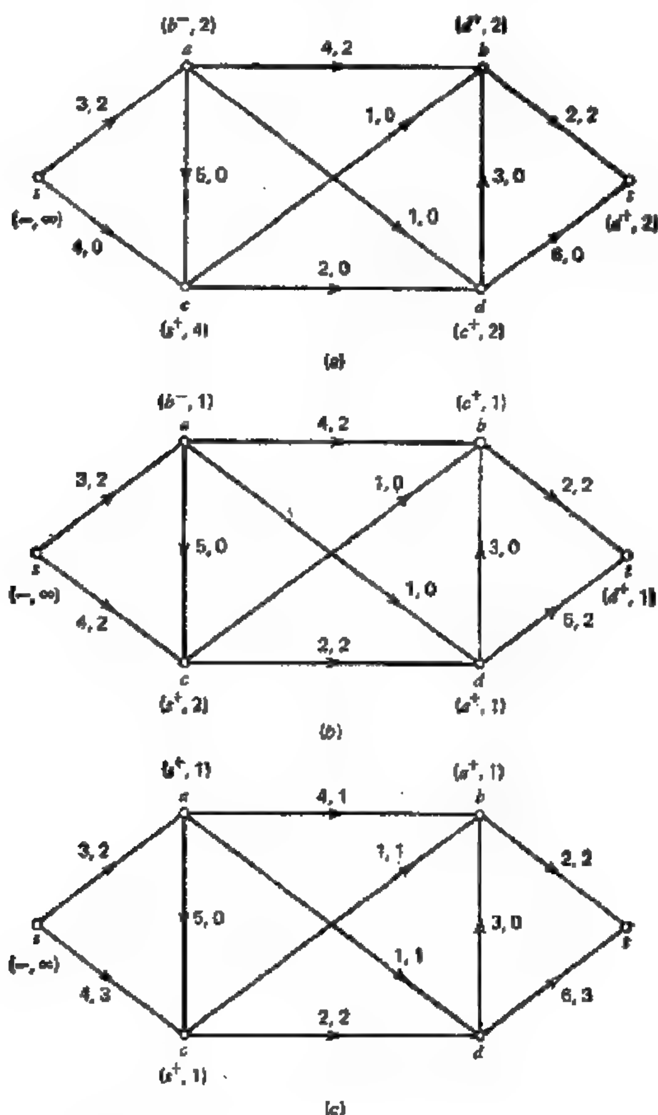


图 15.20 Ford-Fulkerson标号算法举例

15.7.3 Edmonds和Karp对标号算法的改进

在上小节所述的Ford-Fulkerson算法中，顶点可按任意顺序来标记，即可增路径（当它存在时）的选择可以任意方式进行。下面举例说明由于这种任意性可能导致的问题。

考虑图15.21所示的运输网络N。假定我们从零值流开始标号算法，并将路径P_1：s、a、b、t和P_2：s、b、a、t交替用作可增路径，那么流值每步都正好增加1，在增加$2M$步后方得最大流$2M$。这种情况下，计算步数不受N的边数和顶点数的函数来限制。实际上，此步数变为容量M的函数，而M是可以任意大的。

Ford和Fulkerson在文献[15.1]中也进一步说明：当容量是无理数时，其算法就不适用了。在他们给出的一个例子中，流值以无限多步数收敛于最大流值的四分之一。

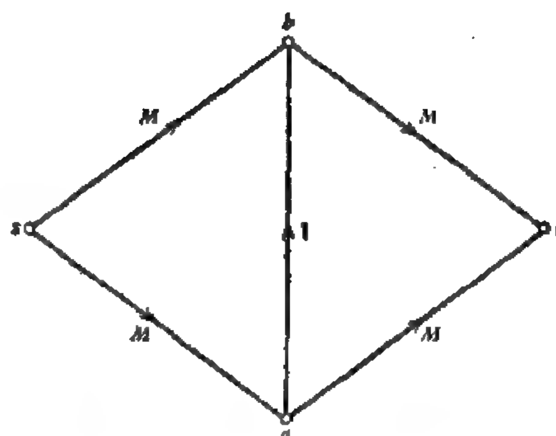


图 15.21

为了避免这种情况，Edmonds和Karp在文献[15.4]中对标号算法提出改进：在每一步，沿着一条最短路径来增广流。此处的最短路径，指的是具有最少边数的路径。显而易见，如果我们在标记过程中以"先标记先检查"为宗旨进行搜索，即可选出一条最短的可增路径，也就是说，如果顶点v在顶点u之前被标号，那么v就在u之前检查。这里检查一个已标号的顶点v就意味着：标记（只要可能）所有与v邻接而尚未标记的顶点。

Edmonds和Karp进一步说明：这一改进将确保需要实现标号算法的计算步数与容量无关。下面讨论这一结果。

考虑运输网络N的流f。令

$$P:\ s=u_0 \xrightarrow{e_1} u_1 \xrightarrow{e_2} u_2 \cdots u_{i-1} \xrightarrow{e_i} u_i \cdots u_{k-1} \xrightarrow{e_k} u_k=t$$

是一条可增路径。曾记得

$$\varepsilon_i(P)=\begin{cases} c(e_i)-f(e_i), & \text{若}e_i\text{是前向边} \\ f(e_i), & \text{若}e_i\text{是后向边} \end{cases}$$

进而

$$\varepsilon(P)=\min_i\{\varepsilon_i(P)\}$$

这样，对某个i，就有$\varepsilon(P)=\varepsilon_i(P)$。对应的$e_i$则称为瓶颈。

假定标号算法从初始流f_0开始，形成序列流f_1，f_2，f_3，…。

我们看到，当e是一条可增路径上前向的瓶颈时，它就在增值过程中达到饱和；如果e是后向的瓶颈，那么在增值过程中，经过e的流就被减少到零。这一观察导致如下结果。

引理15.7　如果$k<p$，以及在把f_k变为f_{k+1}和把f_p变为f_{p+1}的可增路径上，e都是前向（后向）的瓶颈，那么就存在一个l（$k<l<p$），从而在把f_l变为f_{l+1}的可增路径上，e被用作后向（前向）边。□

令$\lambda^l(u,v)$表示从u到v最短的f_l-非饱和路径的长度。当然，此处路径不一定是有向的。又注意到如果边e是非饱和的，即$f_l(e)<c(e)$，那么边e就用作为路径上的前向边，而仅当$f_l(e)>0$时，则边e就用作为后向边。

引理15.8　对每一顶点v和每一$k=0$、1、2、…，

(15.42)

(15.43)

仅证明式(15.42)，而式(15.43)的证明类似。

设不存在从s到v的f_{k+1}-非饱和路径，则$\lambda^{k+1}(s, v)$可认为是无穷大，则(15.42)自然成立。因而假定

$$P:\quad s=u_0 \overset{e_1}{—} u_1 \overset{e_2}{—} u_2 \overset{e_3}{—} u_3 \cdots u_{p-1} \overset{e_p}{—} u_p=v$$

是一条从s到v最短的f_{k+1}-非饱和路径。

设e_i是p的前向边，则$f(e_i)<c(e_i)$。因此，或者(1)$f_k(e_i)<c(e_i)$，或者(2)$f_k(e_i)=c(e_i)$，并且在把f_k变为f_{k+1}的可增路径上，e_i都是后向边。

在前一情形，

$$\lambda^k(s, u_i)\leqslant\lambda^k(s, u_{i-1})+1 \qquad (15.44)$$

这是因为前向的e_i紧跟着从s到u_{i-1}的一条最短的f_k-可增路径，它是从s到u_i的一条f_k-非饱和路径。显然在后一情形，

$$\lambda^k(s, u_{i-1})=\lambda^k(s, u_i)+1 \qquad (15.45)$$

在两种情形下，式(15.44)均成立。

类似地，我们可以证明：当e_i是P上的后向边时，式(15.44)亦成立。

现将式(15.44)对$i=1, 2, \cdots, p$求和，并注意到$\lambda^k(s, u_0)=0$，则有

$$\lambda^k(s, u_p)\leqslant p=\lambda^{k+1}(s, v)$$

即可得到式(15.42) □

引理15.9 如果采用"先标记先检查原则"，并当$k<l$时，l为把f_k变为f_{k+1}的可增路径上的前向(后向)边，并且e为把f_l变为f_{l+1}的可增路径上的后向(前向)边，那么，

$$\lambda^l(s, t)\geqslant\lambda^k(s, t)+2$$

证明

设e从u指向v。那么

$$\lambda^k(s, v)=\lambda^k(s, u)+1 \qquad (15.46)$$

这是因为在增加f_k时，e是前向边。又因为在增加f_l时，e是后向边，所以

$$\lambda^l(s, t)=\lambda^l(s, v)+1+\lambda^l(u, t) \qquad (15.47)$$

但是

$$\lambda^l(s, v)\geqslant\lambda^k(s, v)$$

且

$$\lambda^l(u, t)\geqslant\lambda^k(u, t) \qquad (15.48)$$

由式(15.46)，(15.47)和(15.48)，则得

$$\lambda^l(s, t)\geqslant\lambda^k(s, u)+\lambda^k(u, t)+2$$
$$=\lambda^k(s, t)+2$$

□

定理15.10（Edmonds和Karp） 在Ford-Fulkerson算法中，如果每次增流都是沿着一条最短的可增路经进行，那么获得最大流所需增值次数不超过$\frac{m(n+2)}{2}$，这里，m和n分别是运输网络中的边数和顶点数。

证明

考虑从u指向v的任一边e和流序列f_{k_1}，$f_{k_2}\cdots$，这里$k_1<k_2<\cdots$，并使e在增值f_{k_i}时既是前向边又是瓶颈。根据引理15.7，存在一序列l_1，l_2，…，并满足

$$k_1<l_1<k_2<l_2<k_3<\cdots$$

并且在增值f_{l_i}时，e当作后向边。

由引理15.8，有

$$\lambda^{k_i}(s,\ t)+2\leqslant\lambda^{l_i}(s,\ t)$$

且

$$\lambda^{l_i}(s,\ t)+2\leqslant\lambda^{k_{i+1}}(s,\ t)$$

所以

$$\lambda^{k_1}(s,\ t)+4(j-1)\leqslant\lambda^{k_j}(s,\ t)$$

因为

$$\lambda^{k_j}(s,\ t)\leqslant n-1$$

以及

$$\lambda^{k_1}(s,\ t)\geqslant 1$$

所以得

$$1+4(j-1)\leqslant n-1$$

或

$$j\leqslant\frac{n+2}{4}$$

因此，e用作前向瓶颈的次数最多为$\frac{n+2}{4}$次。同样，e用作后向瓶颈的次数最多也为$\frac{n+2}{4}$次。因此，每条边最多可用作瓶颈的次数为$\frac{n+2}{2}$次。所以，增值的总次数最多为$\frac{m(n+2)}{2}$次。 □

在上述定理的证明中，除了容量必须是非负的以外，再没有做其它假定。根据定理，如果采用“先标记先检查”的原则，那么对于任何非负实容量，在有限次增值后，算法15.9就都能可以结束。因为一条可增路径能在$O(m)$步内找到，那么由定理15.10可知，算法15.9的复杂性为$O(m^2n)$。

Zadeh在文献[15.47]中描绘了这样一类网络：对于此类网络，如果每次增值都是沿最短可增路径进行，那么增值就需要$O(n^3)$次。所以，定理15.10的上限，除了线性算子之外都无法改进。

Dinic在文献[15.48]中证明了：如何利用一次标记过程来求得全部最短可增路径，从

而改进算法15.9的有效性。这一思路与Hopcroft和Karp在文献[15.36]中如何在一个二分图中构造最大匹配所用的方法类似。Dinic算法的复杂性为$O(mn^2)$。

最近，文献[15.49]给出了一个$O(n^3)$算法。后来，在文献[15.50]中又提出了较为简单的$O(n^3)$算法，这个算法与Dinic算法均在文献[15.51]中有所叙述。

文献[15.37]对于特殊类型的运输网络，研究了Dinic算法的复杂性。正如我们早在15.5节中所提到的那样，他们利用这一结果来说明：二分图的最大匹配可在$O(n^{5/2})$时间内构成。

15.7.4 Menger定理重述

本小节，我们将应用最大流最小切割定理来证明对于有向图和无向图的Menger定理（包括定理15.12到15.15）。回忆一下，我们早些时候只对无向图陈述了Menger定理，并未给出证明（见定理8.9及8.12）。在下面的讨论中，为一般性起见，允许源有非零入度，汇有非零出度。

本小节关于Menger定不理的证明基于以下结果。

定理15.11 设N为一具有源s和汇t的运输网络，其中每条边都具有单位容量。那么

1. N中最大流的值等于N中边不相交的有向$s-t$路径的最大数目r。

2. N的最小切割容量等于去掉N中全部有向$s-t$路径所应移去边的最小数目q。

证明

1. 设f^*是N中一个最大流，N^*是移去N的全部f^*-零值边而得到的有向图。因为每条边的容量是单位值，所以对于N^*中的每条边e，显然有$f^*(e)=1$。因此，

a. $d^+_{N^*}(s)-d^-_{N^*}(s)=\text{val}(f^*)=d^-_{N^*}(t)-d^+_{N^*}(t)$。

b. $d^+_{N^*}(v)=d^-_{N^*}(v)$，对所有$v\neq s, t$。

这里$d^+_{N^*}(x)$和$d^-_{N^*}(x)$分别表示在图N^*里顶点x的出度和入度。

于是（练习5.8）N^*中存在$\text{val}(f^*)$条边不相交的有向$s-t$路径，N中亦然。故

$$\text{val}(f^*)\leqslant r \tag{15.49}$$

现在，令$\{P_1, P_2, \cdots, P_r\}$是$N$中$r$条边不相交的有向$s-t$路径的集合，流$f$的定义应满足：

$$f(e)=\begin{cases}1, & \text{若}e\text{属于某}P_i\\ 0, & \text{其它}\end{cases}$$

显然，

$$\text{val}(f)=r$$

因为f^*是一个最大流，所以有

$$\text{val}(f^*)\geqslant r \tag{15.50}$$

综合式（15.49）和（15.50），则得

$$\text{val}(f^*)=r$$

2. 设$K^*=\langle S, \bar{S}\rangle$是$N$中的最小$s-t$切割。假设从$N$中移去边集$\langle S,\bar{S}\rangle$，那么在余下的有向图中不存在有向$s-t$路径。所以，

$$\mathrm{cap}(K^*) = |(S, \bar{S})| \geqslant q \qquad (15.51)$$

再设Z表示一个，移去N中全部有向s-t路径的q条边的集合，并设S表示从s出发的有向路径可到达的所有顶点的集合，其中的有向路径不包含Z中的边。很显然，$K=\langle S, \bar{S}\rangle$是$N$的一个$s-t$切割，进而$\langle S, \bar{S}\rangle \subseteq Z$。所以，

$$\mathrm{cap}(k^*) \leqslant \mathrm{cap}(k) = |(S, \bar{S})| \leqslant |Z| = q \qquad (15.52)$$

综合式（15.51）和（15.52），则得

$$\mathrm{cap}(k^*) = q$$ □

定理15.12 设s和t是有向图G中的两个顶点。那么G中边不相交的有向$s-t$路径的最大数目等于G中移去全部有向$s-t$路径所应移去的边的最小数目。

证明

由G构造一个以s为源、t为汇的运输网络N，并给G的每条边赋予单位容量，那么由定理15.11和最大流最小切割定理即可推导出本定理。 □

对于无向图G，定义一有向图$D(G)$，$D(G)$是通过用一对具有与e相同的端点但互为反向的边来置换边e而得到的。这很易于证明：

1.在G的路径和$D(G)$的有向路径之间存在一一对应关系。

2.对于任意两顶点s和t，G中破坏全部$s-t$路径所应移去的边的最小数目等于$D(G)$中破坏全部有向$s-t$路径所应移去的边的最小数目。

根据上面结果描述对应无向图定理15.12形式的。

定理15.13 设s和t是无向图G的两个顶点。则G中边不相交的$s-t$路径的最大数目等于G中破坏全部$s-t$路径所应移去的边的最小数目。 □

定理15.12和15.13的顶点类比型证明如下。

令s和t是有向图$G=(V, E)$中任意两个不相邻的顶点，依据G按以下方式构造一个有向图G'：

1.把顶点$v \in V-\{s,t\}$分裂成两个新顶点v'和v''，并用一条有向边(v', v'')连接起来。

2.把G的每条以$v \in V-\{s,t\}$为终点的边换成一条新的以v'为终点的边。

3.把G的每条以$v \in V-\{s,t\}$为起点的边换成一条新的以v''为起点的边。

图G以及对应的图G'如图15.22所示。定理不难证明如下：

1.G'中每条有向$s-t$路径都与G中一条由全部缩减为形如(v', v'')的边所得到的有向$s-t$路径相对应。反之，G中每条有向$s-t$路径都与G'中一条除s、t以外的全部在分裂路径上的顶点所得到的有向$s-t$路径相对应。

2.G'中两条有向$s-t$路径是边不相交的，当且仅当G中的对应路径是顶点不相交的。

3.G'中边不相交的有向$s-t$路径的最大数目等于G中顶点不相交的有向$s-t$路径的最大数目。

4.G'中破坏全部有向$s-t$路径所应移去的边的最小数目等于G中破坏全部有向$s-t$路径所应移去的顶点的最小数目。

据此，我们可以得到关于定理15.12的顶点类比表达形式。

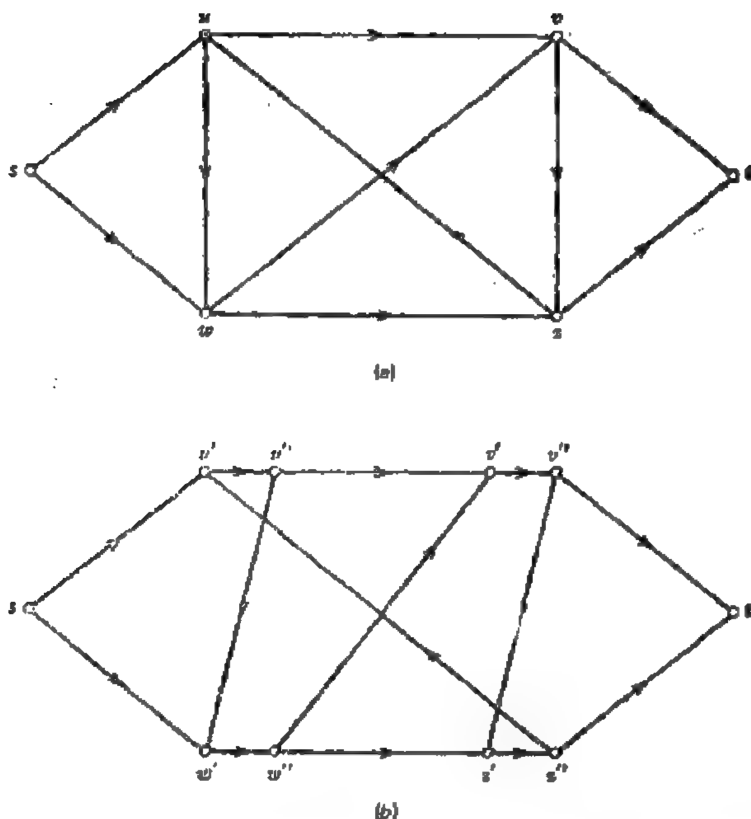


图 15.22 (a)图G; (b)图G'。

定理15.14 设s和t是有向图G中两个不相邻的顶点。那么G中顶点不相交的有向$s-t$路径的最大数目等于G中破坏全部有向$s-t$路径所应移去的顶点的最小数目。 □

关于定理15.13的顶点类比表达形式如下。

定理15.15 设s和t是无向图G中两个不相邻的顶点。那么G中顶点不相交的$s-t$路径的最大数目等于G中破坏全部$s-t$路径所应移去的顶点的最小数目。

证明

对图$D(G)$应用定理15.14，即可证明。 □

有关网络流理论的权威性参考文献是文献[15.1]，那里还讨论了最小费用流问题。读者还可参阅文献[15.46]，[15.52]，[15.8]，[15.53]，[15.7]，[15.9]。

15.8 最优分支

考虑一赋权有向图$G=(V, E)$，设$w(e)$是边e的权。我们定义G的一个子图的权等于该子图中所有边权之和。

如果G的一个子图G_s不含有向回路，且G_s的每个顶点的入度最多为1，那么G_s就称为G的一个分支，很显然，G_s的每个连通片都是一个有向树。具有最大权的分支称为最优分支。

本节，我们讨论由Edmonds在文献[15.54]中提出的计算G的最优分支的算法，讨论

的依据是文献[15.55]。

从顶点 i 指向顶点 j 的边 $e=(i,j)$是关键的，如果

1.$w(e)>0$；

2.对于每条射入 j 的边$e'=(k,j)$，$w(e)\geqslant w(e')$。

G的一个生成子图H是G的一个关键子图，如果

1.H的每条边都是关键的；

2.H中每个顶点的入度最多为1。

有向图G及其关键子图，如图15.23所示。显而易见，

1.关键子图的每个连通片最多含有一个回路，且为一个有向回路。

2.无回路的关键子图是G的一个最优分支。

考虑分支B，设$e=(i,j)$不是B中的边，e'是B中射入到顶点 j 的边，那么 e 对于B是适宜的，如果

$$B'=(B\cup e)-e'$$

是一个分支。

例如，图15.23所示图的边集$\{e_1,\ e_2,\ e_3,\ e_4,\ e_7,\ e_8\}$形成一个分支B。因为

$$(B\cup e_6)-e_7$$

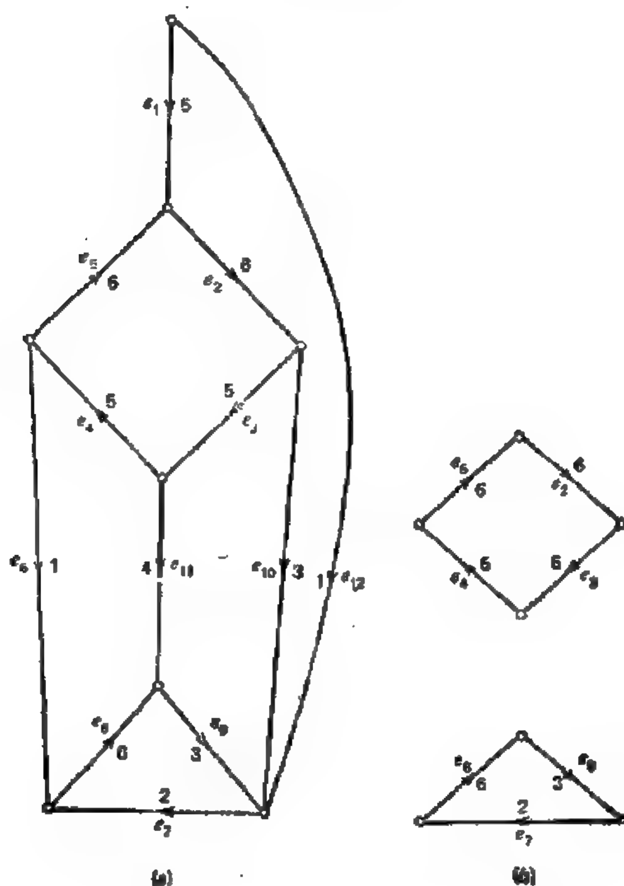


图 15.23 (a)有向图G；(b)G的一个关键子图。

是这个图的一个分支，所以不在B中的边e_6对于B是适宜的。

下面的两个引理很容易证明，由它们可以推出定理15.16；该定理为Karp证明Edmond算法的正确性提供了依据。在以后讨论中，子图H的边集也将用H表示。

引理15.10 设B是一个分支，$e=(i,j)$不是B中的边，那么e对于B是适宜的，当且仅当B中不存在从j到i的有向回路。 □

引理15.11 设B是一个分支，C为一有向回路，并且设$C-B$的边没有一条对于B是适宜的。那么$|C-B|=1$。 □

定理15.16 设H是一个关键子图。那么存在一最优分支B，并使得H的每个有向回路C都满足$|C-B|=1$。

证明

设B为一最优分支，并且在所有最优分支中，B所含关键子图的边数最多。

考虑任一边$e\in H-B$。设e射入到顶点j，e'是B中射入到顶点j的边。如果e是适宜的，那么

$$(B\cup e)-e'$$

也将是一个最优分支，它所含H的边数比B所含的更多，这与假设相矛盾。所以，$H-B$中没有边对于B是适宜的。故由引理15.11可知，对于H中的每个回路C，$|C-B|=1$。

□

设C_1，$C_2\cdots C_k$是H中的有向回路。注意：H中的任何两回路间均无公共边，即这些回路是边不相交的。对于每个C_i，令e_i^0是C_i中一条最小权的边。

推论15.16.1 存在一最优分支B，并使得

1. $|C_i-B|=1$，$i=1,2,\cdots,k$；
2. 如果在$B-C_i$中，没有边与C_i的顶点相关联，$i=1,2,\cdots,k$，那么

$$C_i-B=e_i^0 \tag{15.53}$$

证明

在所有满足（1）的最优分支中，令B是含有集合$\{e_1^0, e_2^0, \cdots, e_k^0\}$中最少边数的分支。现在证明：$B$满足（2）。

如果B不满足(2)，则对于某个i，假定有$e_i^0\in B$，而$B-C_i$的都不射入C_i的顶点。令$e=C_i-B$。那么$(B-e_i^0)\cup e$显然是满足（1）的一个最优分支，但比B所含$\{e_1^0,e_2^0,\cdots,e_k^0\}$的边数更少。这与假设相矛盾。 □

上述结果对于推广Edmonds算法是非常重要的，它使我们有可能仅在满足式(15.33)的分支中寻找最优分支。

现在我们由给定图G来构造一个较为简单的图G'，并说明如何由G'的一个最优分支来构造图G中满足式（15.33）的一个最优分支。

如前所述，设H是G的关键子图，C_1、C_2、…、C_k是H中的有向回路。图G是通过缩减每个C_i（$i=1,2,\cdots,k$）的所有边所形成的。在G'中，每个回路C_i的顶点都是用称之为假顶点的。单个顶点a_i来表示。G'的边权与G中除了射入到假顶点的边之外的边权相同。而那些边权变更如下：

设$e=(i, j)$是G的一条边，并满足j是某回路C_r的一个顶点，而i不在C_r中。那么在G'中，e是入射到假顶点a_r的。定义$\tilde{e}$是C_r中射入到j的唯一边。那么在G'中，e的权（由$w'(e)$表示）为

$$w'(e)=w(e)-w(\tilde{e})+w(e_r^*) \qquad (15.54)$$

例如，在图15.23中G的关键子图中，考虑射入到有向回路$\{e_2, e_3, e_4, e_5\}$的边e_1。那么$\tilde{e}_1=e_5$，且e_1在G'中的权为

$$\begin{aligned} w'(e_1)&=w(e_1)-w(e_5)+w(e_4)\\ &=5-6+5\\ &=4 \end{aligned}$$

注意，e_4是回路$\{e_2, e_3, e_4, e_5\}$中的最小权边。

设E和E'分别为G和G'的边集。现在我们来说明：如何从G'的一个分支B'来构造G的一个满足式（15.53）的分支B，反之亦然。

对于G的满足式（15.53）的任一分支B，显然有

$$B'=B\cap E' \qquad (15.55)$$

B'是G'的一个分支。进而，对于给定的B，上面定义的B'是唯一的。

再考虑G'的一个分支B'。对于每个C_i，按以下方式来定义C_i'：

1. 如果B'中一个假顶点a_i的入度为零，那么

$$C_i'=C_i-e_i^*$$

2. 如果B'中a_i的入度不为零，且e是B'中入射到a_i的边，那么

$$C_i'=C_i-\tilde{e}$$

因此，显而易见

$$B=B'\bigcup_{i=1}^{k}C_i' \qquad (15.56)$$

是G中满足式（15.53）的一个分支，并且对于一给定的B'，上面定义的B是唯一的。

这样，我们就得出结论：满足式（15.53）的G的分支集与G'的分支集之间是一一对应的。进而，对应于分支B与B'的权满足：

$$w(B)-w(B')=\sum_{i=1}^{k}w(C_i)-\sum_{i=1}^{k}w(e_i^*) \qquad (15.57)$$

B与B'的这一性质意味着：如果B是G中满足式（15.53）的一个最优分支，那么B'也是G'的一个最优分支，反之亦然。至此，我们已证明了下面的定理。

定理15.17 G中满足式（15.53）的全部最优分支集与G'的全部最优分支集之间存在一一对应关系。 □

Edmonds的构造最优分支的算法是以上述定理为基础的，现叙述如下：

算法15.10 最优分支（Edmonds）

S1 由给定图$G=G_0$来构造图的一个序列$G_0, G_1, \cdots, G_k$，这里

1. G_k是序列中第一个这样的图，它的关键子图不含回路；

2. $G_i(1\leqslant i\leqslant k)$由$G_{i-1}$得到，其办法是缩减$G_{i-1}$的关键子图$H_{i-1}$的回路，并按式

（15.54）中的要求修改边权。

S2 因为H_k不含回路，所以H_k就是G_k中的一个最优分支。设$B_k=H_k$，构造序列B_{k-1}，B_{k-2}，…，B_0，其中

1.$B_i(0\leqslant i\leqslant k-1)$是$G_i$的一个最优点支；

2.对于$i\geqslant 0$，按式（15.56）扩充B_{i+1}中的假顶点来构造B_i。 □

举例，设图15.23(*a*)中的图为G_0，图15.23(*b*)中的图为H_0，H_0是G_0的关键子图。在缩减了H_0里的回路并改变其边权之后，得到图15.24(*a*)中所示的G_1。G_1的关键子图H_1，如图15.24(*b*)所示。H_1不含回路。因此，H_1是G_1的一个最优分支。再扩充H_1的假顶点a_1和a_2后（它们对应H_0的两个有向回路），就得到G_0的一个最优分支，如图15.24(c)所示。

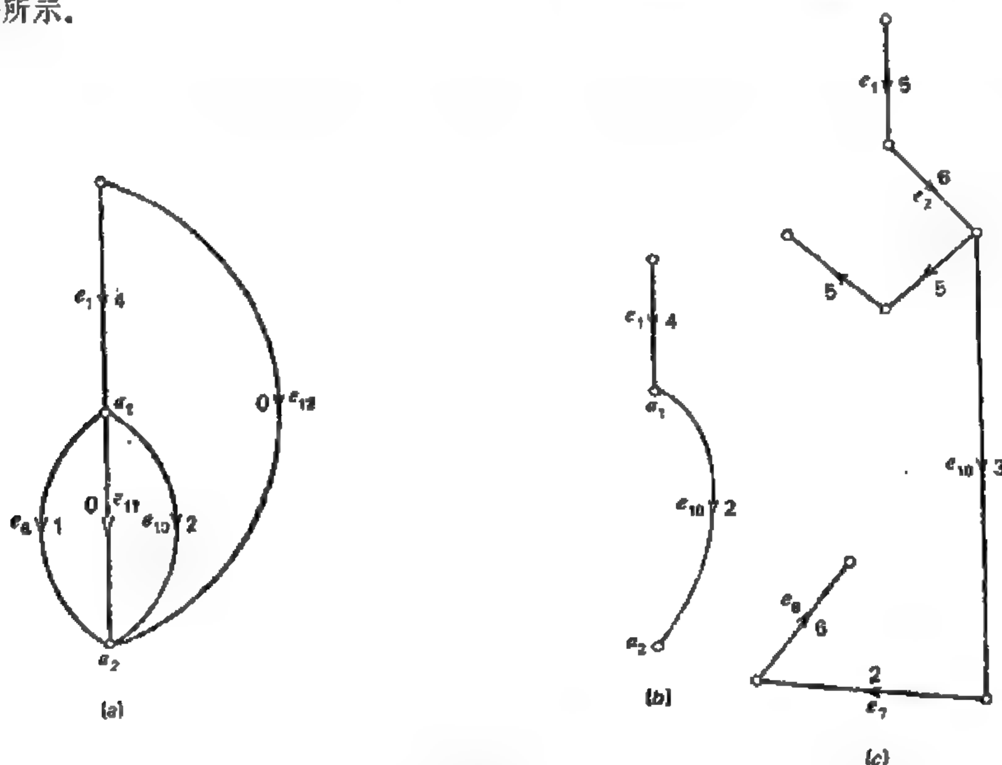


图 15.24 (*a*)图G_1；(*b*)G_1的关键子图H_1；(*c*)图15.23(*a*)中图G的一个最优分支。

文献[15.56]给出了Edmonds算法的一个$O(m\log n)$实现，其中m为边数，n为顶点数。也可参见文献[15.57]，[15.58]，其作者都独立地发现了Edmonds算法。

15.9 进一步阅读

文献著作中已经发表了许多有关图论问题的有趣算法，下面列出其中部分问题，并附有关参考文献。

同构 [15.59]，[15.60]，[15.61]，[15.62]。

平面性 [15.63]，[15.64]，[15.26]，[15.51]。

连通度 [15.65]，[15.66]，[15.37]，[15.67]，[15.51]。

色数 [15.68]，[15.69]。

图的着色 [15.70], [15.71], [15.72], [15.8], [15.73]。

独立集合与稳定性数 [15.74], [15.75], [15.76], 15.77], [15.78], [15.79]。

最小生成树 [15.80], [15.81], [15.82], [15.83], [15.84], [15.10],[15.85], 也参见本书第一部分中的10.9节。

扩张问题 [15.86], [15.87]。

并行计算 [15.88]。

分支 [15.89], [15.90], [15.91], [15.92]。

网络优化 [15.93]。

排进度 [15.94], [15.95], [15.96], [15.97], [15.98], [15.99], [15.100]。

用图论研究线性方程组 [15.101], [15.102], [15.103], [15.104]。

近似算法 [15.105], [15.106]。

15.10 习　　题

15.1 设加权有向图G没有负长度回路。试证：找出特定点s到其它所有点的最短路径的修改Dijkstra算法的有效性。在下面算法步骤中，E代表G的边集，$w(u, v)$表示有向边(u, v)的长度。

S1 对于$u \neq s$，令$\sigma(s)=0$，$\sigma(u)=\infty$。

S2 置$S=\{s\}$。

S3 如果$S=\varnothing$，则停机；否则，选取顶点u^*并使得$\sigma(u^*)=\min_{u \in S}\{\sigma(u)\}$且$u^* \in S$。

S4 对于每个使$(u^*, v) \in E$的v，置$\sigma(v)=\min\{\sigma(v), \sigma(u^*)+w(u^*, v)\}$。若此步能减少$\sigma(v)$，则置$S=S \cup \{v\}$。

S5 置$S=S-\{u^*\}$，并转到S3。

（见文献[15.4]）。

15.2 对于在无负长度有向回路的加权图中，求所有顶点对间最短路的算法，试证：下述由Dantizig在文献[15.10]中给出其算法的有效性。$W^k(i, j)$是从i到j的距离，这里$1 \leqslant i, j \leqslant k$，其路径上没有一个顶点的标号高于$k$。$w(i, j)=W^1(i, j)$是边$(i, j)$的长度，如果不存在这条边，$w(i, j)=\infty$。此外对于所有$i$，$w(i, i)=0$。

S1 置$k=2$。

S2 对于$1 \leqslant i < k$，使

$$W^k(i, k)=\min_{1 \leqslant j < k}\{w(i, k), W^{k-1}(i, j)+w(j, k)\},$$

$$W^k(k, i)=\min_{1 \leqslant j < k}\{w(k, i), w(k, j)+W^{k-1}(j, i)\}。$$

S3 对于$1 \leqslant i, j < k$，使

$$W^k(i, j)=\min\{W^{k-1}(i, j), W^k(i, k)+W^k(k, j)\}。$$

S4 如果$k=n$，停止；否则，置$k=k+1$，并转到S2。在执行上述算法时，也说明了怎样查出负长度回路。

15.3 用拓扑分类的方法，设计出求非周期加权有向图中从顶点s到其它所有顶点的最短路径的算法。

15.4 假定从i到j的最短路径不是唯一的。那么，Floyd算法（算法15.2）将选择哪一条路径？

15.5 证明：如果路径长度矢量（l_1，l_2，…，l_n）满足特征和条件，且$l_1\leqslant l_2\leqslant\cdots\leqslant l_n$及$l_j<l_{j+1}=l_{j+2}=\cdots=l_n$，那么矢量$(l_1, l_2, \cdots, l_j, l_{j+1}-1, l_{j+d+1}, l_{j+d+2}, \cdots, l_n)$也满足特征和条件，路径长度仍按递增顺序排列。（这里d与式（15.17）中定义相同。）

15.6 用习题15.5的结果来说明如何构造m-元树，其中它的树叶从左到右的顺序与路径长度矢量相同。

15.7 对权（1，1，2，3，3，4，4）找出一个最小费用二元搜索树。

15.8 证明：在算法15.5中，将E2用E2′代替。“E2′(选一条边。)选一条边，…，如果这条边不存在，那么转到E1。”
并不影响算法找最大匹配的有效性。（文献[15.28]）

15.9 证明：如果运输网络N中不存在有向$s-t$路径，那么最大流的数值和最小切割的容量都等于零。

15.10 如果$(S, \overline{S})$和$(T, \overline{T})$是运输网络N中的最小切割，证明：$\langle S\cup T, \overline{S\cup T}\rangle$和$\langle S\cap T, \overline{S\cap T}\rangle$也都是$N$中的最小切割。

15.11 证明：在任意具有整数容量的运输网络中，存在一条最大流f，并使得对于N中的每条边e，$f(e)$是整数。

15.12 考虑一个运输网络，其中每个顶点$v\neq s$，t有一个非负整数$m(v)$。说明对改进网络如何应用标号算法找出最大流，并使流入每个顶点$v\neq s$，t的最大流值不超过$m(v)$。（参见建立定理15.14所用的结构。）

15.13 考虑一个运输网络N，其每条边流量的下限已给定。(a)找出N中存在流的充要条件。(b)修改标号算法求出N中的最大流（见文献[15.108]）。

15.14 证明：每条边具有流的下限的运输网络中，存在一个流当且仅当每条边e在有向回路上，或者在$s-t$的有向路径上，或者在$t-s$的有向路径上。

15.15 用找出运输网络N中一条边e的方法，叙述该边具有这种性质，即增加边容量就能增加N中最大流。

15.16 一个最优分支不一定是一个最优有向生成树。说明：怎样修改Edmonds算法，就可求出最优有向生成树。

15.17 (a)证明：下述Prim在文献[15.81]中求加权连通图$G=(V,E)$最小权生成树算法的有效性：
选取G中任一顶点v。在所有与v相关联的边中，选取最小权的边e_1。构造e_1并令G^1是收缩e_1所得到的图。在G^1上继续这个过程，直到e_{n-1}被确定为止。边e_1，e_2，…，e_{n-1}组成了一个G的最小权生成树。
(b)设计一个Prim算法$O(n^2)$的程序。

15.18 用最大流最小切割定理（定理15.9），证明Hall定理（定理8.31），反之亦可证明。

15.19 $G=(X, Y)$是一个二分图，其中从X到Y之间有一个完全匹配。因而可知存在一个顶点$v\in X$，从而对每条与v相关联的边e，都存在一个包含e的完全匹配（见习题8.16）。直接寻找这样的顶点v的算法需要$O(mn^{5/2})$的时间。试找出一个更有效的算法。

15.11 参考文献

15.1 L. R. Ford and D. R. Fulkerson, *Flows in Networks*, Princeton Univ. Press, Princeton, N. J., 1962.

15.2 E. W. Dijkstra, "A Note on Two Problems in Connexion with Graphs," *Numerische Math.*, Vol. 1, 269—271(1959).

15.3 D. B. Johnson, "A Note on Dijkstra's Shortest Path Algorithm," J. *ACM*, Vol. 20, 385—388(1973).

15.4 J. Edmonds and R. M. Karp, "Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems," J. *ACM*, Vol. 19, 248—264(1972)

15.5 S. E. Dreyfus, "An Appraisal of Some Shortest-Path Algorithms," *Operations Research*, Vol. 17, 395—412(1969).

15.6 T. C. Hu, "A Decomposition Algorithm for Shortest Paths in a Network," *Operations Research*, Vol. 16, 91—102(1968)

15.7 H. Frank and I. T. Frisch, *Communication, Transmission and Transportation Networks*, Addison-Wesley, Reading, Mass., 1971.

15.8 N. Christofides, *Graph Theory: An Algorithmic Approach*, Academic Press, New York, 1975.

15.9 E. L. Lawler, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York, 1976.

15.10 P. M. Spira and A. Pan, "On Finding and Updating Spanning Trees and Shortest Paths," *SIAM* J. *Comput.*, Vol. 4, 375—380(1975)

15.11 D. B. Johnson, "Efficient Algorithms for Shortest Paths in Sparse Networks," J. *ACM*, Vol. 24, 1—13(1977).

15.12 R. A. Wagner, "A Shortest Path Algorithm for Edge-Sparse Graphs," J. *ACM*, Vol. 23, 50—57(1976).

15.13 R. W. Floyd, "Algorithm 97: Shortest Path," *Comm. ACM*, Vol. 5, 345(1962).

15.14 Y. Tabourier, "All Shortest Distances., in a Graph: An Improvement to Dantzig's Inductive Algorithm," *Discrete Math.*, Vol. 4, 83—87(1973).

15.15 J. Y. Yen, "Finding the Lengths of All Shortest Paths in *N*-Node, Non-Negative Distance Complete Networks Using $N^3/2$ Additions and N^3 Comparisons," J. *ACM*, Vol. 19, 423—424(1972).

15.16 T. A. Williams and G. P. White, "A Note on Yen's Algorithm for Finding the Length of All Shortest Paths in *N*-Node Non-Negative Distance Networks," J. *ACM*, Vol. 20, 389—390(1973).

15.17 A. R. Pierce, "Bibliography on Algorithms for Shortest Path, Shortest Spanning Tree and Related Circuit Routing Problems(1956-1974)," *Networks*, Vol. 5, 129—149(1975).

15.18 D. A. Huffman, "A Method for the Construction of Minimum Redundancy Codes,"*Proc. IRE*, Vol. 40, 1098—1101(1952).

15.19 E. N. Gilbert and E. F. Moore, "Variable-Length Binary Encodings," *Bell Sys. Tech. J.*, Vol. 38, 933—968(1959).

15.20 D. E. Knuth, "Optimum Binary Search Trees," *Acta Informatica*, Vol. 1, 14—25 (1971).

15.21 A. Itai, "Optimal Alphabetic Trees," *SIAM J. Comput.*, Vol 5, 9—18(1976).
15.22 T. C. Hu and A. C. Tucker, "Optimal Computer Search Trees and Variable-Length Alphabetical Codes," *SIAM J. Appl Math.*, Vol. 21, 514—532(1971).
15.23 T. C. Hu, "A New Proof of the T-C Algorithm," *SIAM J. Appl. Math.*, Vol. 25, 83—94(1973).
15.24 D. E. Knuth, *The Art of Computer Programming, Vol. 3: Sorting and Searching*, Addison-Wesley, Reading, Mass., 1973.
15.25 A. M. Garsia and M. L. Wachs, "A New Algorithm for Minimum Cost Binary Trees," *SIAM J. Comput.*, Vol. 6, 622—642(1977).
15.26 E. M. Reingold, J. Nievergelt, and N. Deo, *Combinatorial Algorithms: Theory and Practice*, Prentice-Hall, Englewood Cliffs, N. J., 1977.
15.27 M. Miyakawa, T. Yuba, Y. Sugito, and M. Hoshi, "Optimum Sequence Trees," *SIAM J. Comput.*, Vol. 6, 201—234(1977).
15.28 J. Edmonds, "Paths, Trees and Flowers," *Canad. J. Math.*, Vol. 17, 449—467 (1965).
15.29 H. N. Gabow, "An Efficient Implementation of Edmonds' Algorithm for Maximum Matching on Graphs," *J. ACM*, Vol. 23, 221—234(1976).
15.30 M. L. Balinski, "Labelling to Obtain a Maximum Matching," in *Combinatorial Mathematics and Its Applications*, (R. C. Bose and T. A. Dowling, Eds.), Univ. North Carolina Press, Chappel Hill, N. C., 1967, pp. 585—602.
15.31 D. Witzgall and C. T. Zahn, Jr., "Modification of Edmonds' Algorithm for Maximum Matching of Graphs," *J. Res. Nat. Bur. Std.*, Vol.69B, 91—98(1965).
15.32 T. Kameda and I. Munro, "A $O(|V|\cdot|E|)$ Algorithm for Maximum Matching of Graphs," *Computing*, Vol 12, 91—98(1974).
15.33 J. Edmonds, "Maximum Matching and a Polyhedron with 0, 1 Vertices," *J. Res. Nat. Bur. Std.*, Vol. 69B, 125—130(1965).
15.34 H. Gabow, "An Efficient Implementation of Edmonds' Maximum Matching Algorithm," Tech. Rep. 31, *Stanford Univ. Comp. Science Dept.*, 1972.
15.35 J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, Macmillan, London, 1976.
15.36 J. E. Hopcroft and R. M. Karp, "An $n^{5/2}$ Algorithm for Maximum Matching in Bipartite Graphs," *SIAM J. Comput.*, Vol. 2,225—231(1973).
15.37 S. Even and R. E. Tarjan, "Network Flow and Testing Graph Connectivity," *SIAM J. Comput.*, Vol. 4, 507—518(1975).
15.38 S. Even and O. Kariv, "An $O(n^{5/2})$ Algorithm for Maximum Matching in General Graphs," *Proc. 16th Annual Symp. on Foundations of Comp. Science*, IEEE, 1975, pp. 100—112.
15.39 A. Itai, M. Rodeh, and S. L. Tanimoto, "Some Matching Problems for Bipartite Graphs," J. *ACM*, Vol. 26, 517—525(1978).
15.40 H. W. Kuhn, "The Hungarian Method for the Assignment Problem," *Naval*

Res. Logist. Quart., Vol. 2, 83—97(1955).

15.4. J. Munkres, "Algorithms for the Assignment and Transportation Problems," *J. SIAM*, Vol. 5, 32—38(1957).

15.42 N. Megido and A. Tamir, "An $O(N\log N)$ Algorithm for a Class of Matching Problems," *SIAM J. Comput.*, Vol. 7, 154—157(1978).

15.43 S. Even, A. Itai, and A Shamir, "On the Complexity of Time-Table and Multicommodity Flow Problems," *SIAM J. Comput.*, Vol. 5, 691—703(1976).

15.44 L. R. Ford and D. R. Fulkerson, "Maximal Flow through a Network," *Canad. J. Math.*, Vol. 8, 399—404(1956).

15.45 P. Elias, A. Feinstein, and C. E. Shannon, "A Note on the Maximum Flow through a Network ," *IRE Trans. Information Theory*, Vol. IT-2, 117—119(1956).

15.46 C.Berge, *Graphs Hypergraphs*, North-Holland, Amsterdam, 1973.

15.47 N. Zadeh, "Theoretical Efficiency of the Edmonds-Karp Algorithm for Computing Maximal Flows," *J. ACM*, Vol. 19, 184—192(1972).

15.48 E. A. Dinic, "Algorithm for the Solution of a Problem of Maximum Flow in a Network with Power Estimation," *Soviet Math. Dokl.*, Vol. 11, 1277—1280(1970).

15.49 A. V. Karzanov, "Determining the Maximal Flow in a Network by the Method of Preflows," *Soviet Math. Dokl.*, Vol 15, 434—437(1974).

15.50 V M. Malhotra, M. Pramodh Kumar, and S. N. Maheswari, "An $O(V^3)$ Algorithm for Maximum Flows in Networks," *Information Processing Lett.*, Vol. 7, 277—278(1978).

15.51 S. Even, *Graph Algorithms*, Computer Science Press, Potomac, Md.,1979.

15.52 C. Berge and A. Ghouila-Houri, *Programming, Games and Transportation Networks*, Wiley, New York, 1962.

15.53 T. C. Hu, *Integer Programming and Network Flows*, Addison-Wesley, Reading, Mass., 1969.

15.54 J. Edmonds, "Optimum Branchings," *J. Res. Nat. Bur. Std.*, Vol. 71B, 233—240 (1967).

15.55 R. M. Karp, "A Simple Derivation of Edmonds' Algorithm for Optimum Branchings," *Networks*, Vol 1, 265—272(1972).

15.56 R. E. Tarjan. "Finding Optimum Branchings," *Networks*, Vol. 7, 25—35(1977).

15.57 Y. Chu and T. Liu, "On the Shortest Arborescence of a Directed Graph," *Scientia Sinica* [*Peking*], Vol. 4, 1396-1400, (1965); *Math. Rev.*, Vol 33, #1245(D. W. Walkup).

15.58 F. C. Bock, "An Algorithm to Construct a Minimum Directed Spanning Tree in a Directed Network," in *Developments in Operations Research*, (B. Avi-Itzak, Ed.), Gordon and Breach, New York, 1971, pp. 29—44.

15.59 D. G. Corneil and C. C. Gotlieb, "An Efficient Algorithm for Graph Isomorphism," *J. ACM*, Vol. 17, 51—64(1970).

15.60 L. Weinberg, "A Simple and Efficient Algorithm for Determining Isomorphism

of Planar Triply Connected Graphs," *IEEE Trans. Circuit Theory*, Vol. CT-13, 142—148(1966).

15.61 J. E. Hopcroft and R. E. Tarjan, "A VlogV Algorithm for Isomorphism of Tri connected Planar Graphs," *J. Comput. Syst. Sci.*, Vol. 7, 323—331(1973).

15.62 J. E. Hopcroft and J. K. Wong, "Linear Time Algorithm for Isomorphism on Planar Graphs," *Proc. 6th Annual ACM Symp. on Theory of Computing*, 1974, pp. 172—184.

15.63 J. E. Hopcroft and R. E. Tarjan, "Efficient Planarity Testing," J. *ACM*, Vol. 21, 549—568(1974).

15.64 N. Deo, "Note on Hopcroft and Tarjan's Planarity Algorithm," *J. ACM*, Vol. 23, 74—75(1976).

15.65 J. E. Hopcroft and R. E. Tarjan, "Dividing a Graph into Triconnected Components," *SIAM J. Comput.*, Vol. 2, 135—158(1973).

15.66 D. J. Kleitman, "Methods for Investigating the Connectivity of Large Graphs," *IEEE Trans. Circuit Theory*, Vol. CT-16, 232—233(1969).

15.67 S. Even, "An Algorithm for Determining whether the Connectivity of a Graph is at Least *k*," *SIAM* J. *Comput.*, Vol. 4, 393—396(1975).

15.68 D. G. Corneil and B. Graham, "An Algortihm for Determining the Chromatic Number of a Graph," *SIAM* J. *Comput.*, Vol. 2, 311—318(1973).

15.69 C. McDiarmid, "Determining the Chromatic Number of a Graph," *SIAM J. Comput.*, Vol. 8, 1—14(1979).

15.70 D. J. A. Welsh and M. B. Powell, "An Upper Bound for the Chromatic Number of a Graph and Its Applications to Timetabling Problems," *The Computer J.*, Vol. 10, 85—86(1967).

15.71 D. C. Wood, "A Technique for Colouring a Graph Applicable to Large Scale Timetabling Problems," *The Computer* J., Vol. 10, 317—319(1969).

15.72 D. Matula, G. Marble, and J. Isaacson, "Graph Colouring Algorithms," in *Graph Theory and Computing*, (R. C. Read, Ed.), Academic Press, New York, 1972, pp. 109—122.

15.73 D. Brélaz, "New Methods to Color the Vertices of a Graph," *Comm. ACM*, Vol. 22, 251—256(1979).

15.74 M. C. Paull and S. H. Unger, "Minimizing the Number of States in Incompletely Specified Sequential Switching Functions," *IRE Trans. Elect Comput.*, Vol. EC-8, 356—357(1959).

15.75 C. Bron and J. Kerbosch, "Finding All Cliques of an Undirected Graph—Algorithm 457," *Comm. ACM*, Vol. 16, 575—577(1973).

15.76 E. A. Akkoyunlu, "The Enumeration of Maximal Cliques of Large Graphs," *SIAM* J. *Comput.*, Vol 2, 1—6(1973).

15.77 S. Tsukiyama, M. Ide, H. Ariyoshi, and I. Shirakawa, "A New Algorithm for Generating All the Maximal Independent Sets," *SIAM* J. *Comput.*, Vol. 6, 505

—517(1977).

15.78 R. E. Tarjan and A. E. Trojanowski, "Finding a Maximum Independent Set," *SIAM J Comput.*, Vol. 6, 537—546(1977).

15.79 V. Chvátal, "Determining the Stability Number of a Graph," *SIAM J Comput.*, Vol. 6, 643—662(1977).

15.80 J. B. Kruakal, Jr., "On the Shortest Spanning Subtree of a Graph and the Travelling Salesman Problem," *Proc. Am. Math. Soc.*, Vol. 7, 48—50(1956)

15.81 R. C. Prim, "Shortest Connection Networks and Some Generalizations," *Bell Sys. Tech. J.*, Vol. 36, 1389—1401(1957).

15.82 A. Kerschenbaum and R. Van Slyke, "Computing Minimum Spanning Trees Efficiently," *Proc. 25th Ann. Conf. of the ACM*, 1972, pp. 518—527.

15.83 A. C. Yao, "An O(|E|loglog|V|) Algorithm for Finding Minimum Spanning Trees," *Information Processing Lett.*, Vol. 4, 21—23(1975).

15.84 D. Cheriton and R. E. Tarjan, "Finding Minimum Spanning Trees," *SIAM J. Comput.*, Vol. 5, 724—742(1976).

15.85 H. N. Gâbow, "Two Algorithms for Generating Spanning Trees in Order," *SIAM J. Comput.*, Vol. 6, 139—150(1977).

15.86 K. P. Eswaran and R. E. Tarjan, "Augmentation Problems," *SIAM J. Comput.*, Vol. 5, 653—665(1976).

15.87 A. Rosenthal and A. Goldner, "Smallest Augmentations to Biconnect a Graph," *SIAM J. Comput.*, Vol. 6,55—66(1977).

15.88 E. Reghbati and D. G. Corneil, "Parallel Computations in Graph Theory," *SIAM J. Comput.*, Vol. 7, 230—237(1978).

15.89 J. Edmonds, "Edge-Disjoint Branchings," in *Combinatorial Algorithms*, (R. Rustin, Ed.), Algorithmics Press, New York, 1973, pp. 91—96.

15.90 D. R. Fulkerson and G. C. Harding, "On Edge-Disjoint Branchings," *Networks*, Vol. 6, 97—104(1976).

15.91 R. E. Tarjan, "Edge-Disjoint Spanning Trees and Depth-First Search," *Acta Informatica*, Vol. 6, 171—185(1976).

15.92 L. Lovász, "On Two Minimax Theorems in Graph," *J. Combinatorial Theory B*, Vol. 21, 96—103(1976),

15.93 B. L. Golden and T. L. Magnanti, "Deterministic Network Optimization: A Bibliography," *Networks*, Vol. 7, 149—183(1977).

15.94 M. Fujii, T. Kasami, and K. Ninomiya, "Optimal Sequencing of Two Equivalent Processors," *SIAM J. Appl. Math.*, Vol. 17, 784—789(1969); Erratum, Vol. 20, 141(1971).

15.95 E. G. Coffman, Jr., and R. L. Graham, "Optimal Scheduling for Two-Processor Systems," *Acta Informatica*, Vol. 1, 200—213(1972).

15.96 E. G. Coffman, Jr., and P. J. Denning, *Operating System Theory*, Prentice Hall, Englewood Cliffs, N. J., 1973.

15.97 R. Sethi, "Scheduling Graphs on Two Processors," *SIAM J. Comput.*, Vol. 5, 73—82(1976)

15.98 R. Sethi, "Algorithms for Minimal Length Schedules," in *Computer and Job Scheduling Theory* (E. G. Coffman, Jr., Ed.), Wiley, New York, 1976, pp. 51—99.

15.99 J. D. Ullman, "Complexity of Sequencing Problems," in *Computer and Job Scheduling Theory*, (E. G. Coffman, Jr., Ed.), Wiley, New York, 1976, pp. 139—164.

15.100 S. Lam and R. Sethi, "Worst Case Analysis of Two Scheduling Algorithms," *SIAM J. Comput.*, Vol. 6, 518—536(1977).

15.101 D. J. Rose, "A Graph-Theoretic Study of Numerical Solutions of Sparse Positive Definite Systems of Linear Equations," in *Graph Theory and Computing* (R. C. Read, Ed.), Academic Press, New York, 1972, pp. 183—217.

15.102 T. Ohtsuki, "A Fast Algorithm for Finding an Optimal Ordering for Vertex Elimination on a Graph," *SIAM J. Comput.*, Vol. 5, 133—145(1976)

15.103 D. J. Rose, R. E. Tarjan, and G. S. Lueker, "Algorithmic Aspects of Vertex Elimination on Graphs," *SIAM J. Comput.*, Vol. 5, 266—283(1976).

15.104 R. E. Tarjan, "Graph Theory and Gauss Elimination," in *Sparse Matrix Computations*, (J. R. Bunch and D. J. Rose, Eds.), Academic Press, New York, 1976.

15.105 E. Horowitz and S. Sahni, *Fundamentals of Computer Algorithms*, Computer Science Press, Potomac, Md., 1978.

15.106 M. R. Garey and D. S. Johnson, "Approximation Algorithms for Combinatorial Problems: An Annotated Bibliography," in *Algorithms and Comlexity: New Directions and Recent Results* (J. Traub, Ed.) Academic Press, New York 1976.

15.107 G. B. Dantzig, "All Shortest Routes in a Graph," in *Theory of Graphs*, Gordon and Breach, New York, 1967, pp. 91—92

15.108 S. Even, *Algorithmic Combinatorics*, Macmillan, New York, 1973.

名 词 索 引

二 画

二元关系 binary relation
二元关系的闭集 closed set relative to binary relation
二元树 binary tree
二分拟阵 bipartite matroid
二分图 bipartite graph
二划分 bipartition
二连通片 biconnectivity component
入度矩阵 indegree matrix

三 画

大子式 major(determinant)
子图 subgraph
　k-子图 k-subgraph
　主子图 principal subgraph
　生成子图 spanning subgraph
　导出子图 induced subgraph
　匈牙利子图 Hungarian subgraph
　极大子图 maximal subgraph
　极小子图 minimal subgraph
　真子图 proper subgraph
　临界子图 critical subgraph
　等价子图 equality subgraph
子图的权积 weight product of subgraph
子图的补图 complement of subgraph
子图的直和 direct sum of subgraph
子空间 subspace
　正交子空间 orthogonal subspace
　回路子空间 circuit subspace
　割集子空间 cutset subspace
子树 subtree
　左子树 left subtree
　右子树 right subtree
子集 subset
　极大子集 maximal subset
　极小子集 minimal subset

四 画

匹配 matching
　完全匹配 complete matching
　完美匹配 perfect matching
　最大匹配 maximum matching
　最优匹配 optimal matching
匹配拟阵 matching matroid
匹配数 matching number
不同代表系统 system of distinct representives
不定导纳矩阵 indefinite admittance matrix
无向图全部点关联矩阵 all vertices incidence matrix of undirected graph
无增益性 no-gain property
双重随机矩阵 doubly stochastic matrix
开路阻抗矩阵 open-circuit impedance matrix
反向标号 backward labeling
反射关系 reflexive relation
反射有向图 reflexive directed graph
反馈环 feedback loop
分支 branching
　最优分支 optimum branching
　最优分支算法 optimum branching algorithm
分割 seg
分配律 commutative property
比分 score
比分序列 score sequence
片 component(s)
　二连通片 biconnected component
　有向图的片 component of dire-

cted graph
奇片 odd component
偶片 even component
图的片 component of graph
切割 cut
有向切割 directed cut
最小切割 minimum cut
切割矢量 cut vector
切割容量 capacity of cut
切割矩阵 cut matrix
有向图的切割矩阵 cut matrix of directed graph
图的切割矩阵 cut matrix of graph

五　画

半强连通图 quasi-strongly connected graph
汇点 sink
节点 node
节点方程 node equations
节点电压 node voltage
节点导纳矩阵 node admittance matrix
节点变换 node transformation
节点参考电压 node-to-datum voltage
平行边 parallel edge
平行计算 parallel computation
平面图 planar graph
平面图的域 region of planar graph
平面网络 planar network
对偶平面网络 dual of planar network
平面单端对图 planar one-treminal-pair graph
平面嵌入 planar embedding
边 edge
k-边有向连通图 k-edge-connected directed graph
k-边连通图 k-edge connected graph
k-边着色 k-edge colouring
真k-边着色 proper k-edge colouring
最优k-边着色 optimum k-edge colouring
反向边 reverse edge
f-正向边 f-positive edge
f-未饱和边 f-unsaturated edge
回路边 circuit edge
回溯边 back edge
合适边 eligible edge
邻接边 adjancent edge
非回路边 noncircuit edge
前向边 forward edge
标记边 marked edge
树边 tree edge
悬挂边 pendant edge
端口边 port edge
临界边 critical edge
边的长度 length of edge
边的收缩 contraction of edge
边的容量 capacity of edge
边移去 edge removal
边覆盖 edge cover
最小边覆盖 minimum edge cover
四色问题 four colour problem
四色定理 four colour theorem
四色猜想 four colour conjecture
出度矩阵 out degree matrix
矢量 vector
n-矢量 n-vector
矢量的内积 inner product of vectors
矢量的座标 coordinates of vector
矢量的点积 dot product of vectors
矢量空间 vector space
同构矢量空间 isomorphic vector space
矢量空间的基底 basis of vector space
正交关系 orthogonality relation
正交子空间 orthogonal subspaces
正交矢量 orthogonal vectors
正交互补 orthogonal comple-

	ments
对称关系	symmetric relations
对称有向图	symmetric directed graph
对偶网络	dual networks
可行顶点标号	feasible vertex labeling
生成补树	cospanning tree

六 画

关联矢量	incidence vector
关联矩阵	incidence matrix
关联集	incidence set
交错序列	alternative sequence
交错链	alternative chain
守恒条件	conservation condition
导纳矩阵	admittance matrix
导纳乘积	admittance product
有向切割	directed cut
有向回路	directed circuit
有向开通道	directed open walk
有向闭通道	directed closed walk
有向连通图	connected directed graph
有向图，有向图1-因子连接	directed graph, 1-factorial connection in directed graph
有向图切割矩阵	cut matrix of directed graph
有向图回路矩阵	circuit matrix of directed graph
有向图邻接矩阵	adjacency matrix of directed graph
有向欧拉轨迹	directed Eulerian trail
有向欧拉图	directed Eulerian graph
有向哈密顿图	directed Hamilton graph
有向割集	directed cutset
有向路径	directed path
轨迹	trail
开轨迹	open trail
有向轨迹	directed trail
闭轨迹	closed trail
欧拉轨迹	Euler trail
回路	circuit(s)
有向回路	directed circuit
有向图的回路	circuit in directed graph
有向哈密顿回路	directed Hamilton circuit
哈密顿回路	Hamilton circuit
补回路	cocircuit
基本补回路	fundamental cocircuit
拟阵的回路	circuit in matroid
拟阵的基本回路	fundamental circuit in matroid
图的回路	circuit in graph
基本回路	fundamental circuit
回路子空间	circuit subspace
回路方程组	loop system(of equations)
回路长度	length of circuit
回路矢量	circuit vector
回路变换	circuit transformation
回路变量	loop variable
回路矩阵	circuit matrix
回路基本集合	fundamental set of circuits
回路的权	weight of circuit
回路和割集子空间的正交性	orthogonality of circuit and cutset subspace
因子	factor
k-因子	k-factor
k-因子可分解图	k-factorable graph
有向图的1-因子	1-factor in directed graph
团集	clique
网络优化	network optimization
网络灵敏度计算	network sensitivity computation
网络函数拓扑公式	topological formula for network function .

传递关系	transitive relation
传递有向图	transitively directed graph
传递闭包	transitive closure
二元关系传递闭包	transitive closure of binary relation
图的传递闭包	transitive closure ofgraph
传递可定向图	transitively orientable graph
自对偶图	self-dual graph
自补图	self-complementary graph
自环	self-loop
优化	majorize
价	valency
同构图	isomorphic graphs
1-同构图	1-isomorphic graphs
2-同构图	2-isomorphic graphs
同胚图	homeomorphic graphs
色多项式	chromatic polynomial
色指数	chromatic index
色数	chromatic number
k-色数图	k-chromatic graph

七 画

完备k-分图	complete k-partite graph
完备二分图	complete bipartite graph
完备有向图	complete directed graph
完备图	complete graph
补回路矩阵	cocircuit matrix
补林（树）	coforest(cotree)
补图	cograph
补图拟阵	cographic matroid
状态方程	state equations
运输网络	transport network
极图理论	extremal graph theory
连支（弦）	chord, link
非端口连支	nonport link
端口连支	port link
连通图	connected graph
有向连通图	connected directed graph
最小有向连通图	minimally connected directed graph
最小连通图	minimally connected graph
强连通图	strongly connected graph
准强连通图	quasi-strongly connected graph
k-连通图	k-connected graph
连通度	connectivity
边连通度	edge connectivity
顶点连通度	vertex connectivity
连通点	connected vertex
花	blossom
均匀锥形矩阵	uniformly tapered matrix
拟阵	matroid
Fano拟阵	Fano matroid
二元拟阵	binary matroid
二分拟阵	bipartite matroid
匹配拟阵	matching matroid
正则拟阵	regular matroid
可分拟阵	separable matroid
不可分拟阵	nonseparable matroid
可定向拟阵	orientable matroid
可表示拟阵	representable matroid
对偶拟阵	dual matroid
同构拟阵	isomorphic matroid
回路拟阵	circuit matroid
补图拟阵	cographic matroid
均匀拟阵	uniform matroid
连通拟阵	connected matroid
非连通拟阵	disconnected matroid
欧拉拟阵	Eulerian matroid
图的拟阵	graphic matroid
基可排序拟阵	base orientable matroid
游戏拟阵	gamoid matroid
割集拟阵	cutset matroid
键拟阵	bond matroid

横截拟阵	transversal matroid
拟阵的元素	elements of matroid
拟阵的并	union of matroids
拟阵的闭集	closed set in matroid
拟阵的回路	circuit in matroid
拟阵的回路公理	circuit axioms for matroid
拟阵的回路矩阵	circuit matrix of matroid
拟阵的约束	restriction of matroid
拟阵的补回路	cocircuit in matroid
拟阵的补回路矩阵	cocircuit matrix of matroid
拟阵的补环	coloop in matroid
拟阵的补基	cobase in matroid
拟阵的收缩	contraction of matroid
拟阵的余因子	minor of matroid
拟阵的环	loop in matroid
拟阵的表示	representation of matroid
拟阵的标准表示	standard representation of matroid
拟阵的相关集	dependent set in matroid
拟阵的独立集	independent set in matroid
拟阵的独立性公理	independence axioms for matroid
拟阵的秩	rank of matroid
拟阵的秩公理	rank axims for matroid
拟阵的秩函数	rank function of matroid
拟阵的超平面	hyperplane in matroid
拟阵的基	base in matroid
拟阵的基公理	base axioms for matroid
拟阵的基本回路	fundamental circuit in matroid
拟阵的基本回路矩阵	fundamental circuit matrix of matroid
拟阵的基本补回路	fundamental cocircuit in matroid
拟阵的基本补回路矩阵	fundamental cocircuit matrix of matroid
改进回路矩阵	modified circuit matrix
阻抗乘积	impedance product
阿贝尔群	Abelian group
克希荷夫电压定律	Kirchhoff's voltage law
克希荷夫电流定律	Kirchhoff's current law
直径，围长	diameter, girth
收缩图	condensed graph
收缩映象	condensed image
时间表安排	timetable scheduling
串联插入	series insertion
串联溶合	series merger
邻接边	adjancent edges
邻接表	adjancency list
邻接顶点	adjancent vertices
伴随有向图	associated directed graph
伴随网络	adjoint of a network

八 画

定理	theorem
（交错链的）Berge定理	Berge's(alternative chain)theorem
Brook定理	Brook's theorem
Cayley定理	Cayley's theorem
Edmonds k-边连通度定理	Edmond's theorem on k-edge connectivity
Hall定理	Hall's theorem
König定理	König's theorem
König-Egerváry定理	König-Egerváry's theorem
Kuratowski定理	Kuratowski's theorem
Mendelsohn和Dulmage定理	Mendelsohn and Dulmage's theorem
Menger定理	Menger's theorem
Menger定理的证明	Proof of Menger's

theorem
Tutte完美匹配定理 Tutte's perfect matching theorem
Vizing定理 Vizing's theorem
四色定理 four-colour theorem
五色定理 five colour theorem
比耐-柯西定理 Binet-Cauchy theorem
涂色定理 painting theorem
特勒根定理 Tellegan's theorem
最大流最小切割定理 max-flow min-cut theorem
增广定理 augmentation theorem
单位元素 identity element
单模矩阵 unimodular matrix
林 forest
顶点（点） vertex
子顶点 son of vertex
无关顶点 unrelated vertex
内点 internal vertex
有向通道的内点 internal vertex of directed walk
通道的内点 internal vertex of walk
n-端口网络的内点 internal vertex of n-port network
外点 outer vertex
左顶点 left of vertex
右顶点 right of vertex
叶顶点 leaf of vertex
后继顶点 decendent of vertex
真后继顶点 proper decendent of vertex
连通顶点 connected vertex
强连通顶点 strongly connected vertex
邻接顶点 adjancent vertices
参考顶点 reference vertex
（匹配中的）饱和顶点 saturating vertex (in matching)
孤立顶点 isolated vertex
终点 terminal vertex
边的终点 terminal vertex of edge
有向通道的终点 terminal vertex of directed walk
通道的终点 terminal vertex of walk
始点 initial vertex
边的始点 initial vertex of edge
有向通道的始点 initial vertex of directed walk
起点 start vertex(in program graph)
前代顶点 ancestor of a vertex
真前代顶点 proper ancestor of a vertex
标记顶点 marked vertex
标号顶点 labeled vertex
相关顶点 related vertexes
相邻顶点 adjacent vertexes
星顶点 star vertex
梢顶点 tip vertex
虚拟顶点 pseudo vertex
悬挂顶点 pendent vertex
割点 cut vertex
端点 end vertex
顶点间的距离 distance between vertices
顶点不相交图 vertex-disioint graph
顶点分裂 collapsing a vertex
顶点导出子图 vertex-induced subgraph
顶点连通度 vertex connectivity
顶点的标号 label of a vertex
顶点的度 degree of a vertex
顶点的深度 depth of a vertex
k-顶点着色 k-vertex colouring
真k-顶点着色 proper k-vertex colouring
顶点移去 vertex removal
顶点覆盖 vertex cover
顶点覆盖数 vertex covering number
欧拉公式 Euler's formula
图 graph
Petersen图 Peterson graph
二分图 bipartite graph
无向图 nonoriented(undirected)graph
无圈图 acyclic graph

立方图	cubic graph
对偶图	dual graph
平凡图	trival graph
平面单端对图	planar one-pair-terminal graph
平面图	planar graph
正则图	regular graph
边不相交图	edge-disjoint graph
k-因子化图	k-factorable graph
k-边可着色图	k-edge colourable graph
k-边连通图	k-edge connected graph
k-边有向连通图	k-edge directed connected graph
k-边着色图	k-edge chromatic graph
可分图	separable graph
不可分图	nonseparable graph
可化简程序图	reducible program graph
加权图	weighted graph
比赛图	tournament
k-连通图	k-connected graph
有向无圈图	directed acyclic graph
有向图	directed graph
有向欧拉图	directed Eulerian graph
有向哈密顿图	directed Hamilton graph
完全图	complete graph
完全k-分图	complete k-partite graph
完全二分图	complete biparite graph
完全有向图	complete directed graph
串并图	series-parallel graph
串并图的并行组合	parallel combination of series-parallel graph
串并图的串行组合	series combination of series-parallel graph
空图	empty graph
零图	null graph
单端对图	one-terminal-pair graph
欧拉图	Eulerian graph
轮辐图	wheel graph
顶点不相交图	vertex-disjoint graph
k-顶点可着色图	k-vertex colourable graph
哈密顿图	Hamilton graph
哈密顿连通图	Hamilton connected graph
基础无向图	underlying undirected graph
简单图	simple graph
简单图的补图	complement of simple graph
程序图	program graph
随机欧拉图	randomly Eulerian graph
随机哈密顿图	randomly Hamilton graph
图上运算	operations on graphs
图序列	graph sequence
图阵	garphoid
图的交	intersection of graphs
图的并	union of graphs
图的闭包	closure of graph
图的约束	restriction of graph
图的收缩	contraction of grah
图的环和	ring sum of graphs
图的切割矩阵	cut matrix of graph
图的回路矩阵	circuit matrix of graph
拓扑分类	topological sorting
拓扑自由度	topological degree of freedom
弧着色引理	arc colouring lemma
转置网络	transpose of network
非连通集合	disconnected set
线性独立回路矢量	linearly independent circuit vector

九 画

度	degree

顶点的度　degree of vertex
入度　in-degree
域的度　degree of region
亏效度　deficiency
出度　out-degree
厚度　thickness
度序列　degree sequence
派生序列　derivation sequence
派生链　derivation chain
逆元　inverse of element in a group
前缀码　prefix code
标量　scaler
树　tree
k 树　k-tree
二元树　binary tree
二元搜索树　binary searching tree
最优二元搜索树　optimum binary search tree
增广二元搜索树　augmented binary search tree
正则树　normal tree
生成k-树　spanning k-tree
生成树　spanning tree
生成k-补树　cospanning k-tree
生成补树　cospanning tree
生成树之间的距离　distance between spanning trees
主控顶点树　dominator tree
M-交错树　M-alternating tree
有向生成树　directed spanning tree
有向树（单向树）　directed tree(arborescence)
有向图的树　tree in directed graph
有向入射树　in-going directed tree
有向入射树的根　root of in-going directed tree
有序树　ordered tree
扩展二元搜索树　extended binary searching tree
匈牙利树　Hungarian tree
深度优先搜索树　depth-first searching tree
最大加权生成树　maximum weight spanning tree
最大距离生成树　maximally distant spanning tree
最优二元生成树　optimum binary spanning tree
柯茨图　Coates graph
柯茨流图　Coates flow graph
柯茨方法　Coates method
柯茨增益公式　Coates gain formula

十　画

容量限制　capacity constraint
流　flow
最大流　maximum flow
哥尼斯堡桥问题　Königsberg bridge problem
秩　rank
有向图的秩　rank of directed graph
拟阵的秩　rank of matroid
图的秩　rank of graph
通道　walk
闭通道　closed walk
有向闭通道　directed closed walk
有向通道　directed walk
开通道　open walk
有向开通道　directed open walk
涂色　painting
可定向拟阵的涂色　painting of orientable matroid
涂色定理　painting theorem

十 一 画

深度优先搜索　depth-first search
混合变量系统　mixed-variable system(of equations)
混合变量法　mixed-variable method(of network)
域　field
Galois域　Galois field
排序　ranking
梅森图　Mason graph
梅森信号流图　Mason's signal flow graph

梅森增益公式　Mason's gain formula
梅森方法　Mason's method
控主　dominator
　半控主　semi-dominator
　直接控主　immediate dominator
敏感度　susceptibility

十 二 画

割点　cut vertex
割集　cutset(s)
　有向割集　directed cutset
　基本割集　fundamental cutset
割集方程组　cutset system of equation(s)
割集的基本集合　fundamental set of cutset
割集变换　cutset transformation
割集变量　cutset variable
超平面　hyperplane
最大独立集　maximum independent set
等余因子矩阵　equi-cofactor matrix
等价关系　equivalence relation
等价族　equivalence classes
程序图的可化简性　reducibility of program graph

十 三 画

解码　decode
零度　nullity
　有向图的零度　nullity of directed graph
　图的零度　nullity of graph
零空间　null space
路径　path
　f-未饱和路径　f-unsaturated path
　有向图的路径　path in directed graph
　有向哈密顿路径　directed Hamiltonian path
　有向路径　directed path
　图的路径　path in graph
　哈密顿路径　Hamiltonian path
　前向路　forward path
　最长路径　maximal path
　最短路径　shortest path
　增广路径　augmenting path
　f-增广路　f-augmenting path
路径长度　length of path
路径长度矢量　path length vector
路径长度矢量特征和　characteristic sum of path length vector
路径压缩　path compression
群　group
简化顺序　reduction order

十 四 画

端口　port
　同向端口　similarly oriented ports
　非同向端口　dissimilarly oriented ports
　梢端口　tip port
端口变换　port transformation
端口结构　port configuration
　线性树端口结构　linear tree port configuration
　星形树端口结构　star tree port configuration
算法　algorithm(s)
　Greedy算法　Greedy algorithm
　二连通性算法　biconnectivity algorithm
　广度优先搜索算法　breadth-first search algorithm
　无向图的深度优先搜索算法　depth-first search algorithm of undirected graph
　有向图的深度优先搜索算法　depth-first search algorithm of directed graph
　时间表安排算法　timetable scheduling algorithm
　求一般图的最大匹配　maximum matching in general graph
　求二分图的最大匹配　maximum matching in bipartite graph
　求控主算法　dominators algori-

	thm
求传递闭包算法	transtive closure algorithm
求完美匹配	perfect matching
求运输网络中的最大流	maximum flow in transport network
求给定度数的图	graphs with prescribed degrees
求最小加权生成树	minimum weighted spanning tree
求最优二元搜索树	optimum binary search tree
求最优分支	optimum branching
求最优分配	optimal assignment
求最优路径长度矢量	optimal path length vector algorithm
求最短路径	shortest path
所有顶点对间最短路	shortest paths between all pairs of vertices
从指定点到其它所有顶点间最短路径	shortest paths from specified vertex to all vertices
近似算法	approximation algorithm
Edmonds和Karp的改进标号算法	Edmonds and Karp's modification of labeling algorithm
实现(n+1)节点的n端口电阻网络	realization of(n+1) node resistance network
实现割集/回路矩阵	realization of cutset /circuit matrix
检查传递定向算法	transitive orientation algorithm
检查程序图的可约性算法	program graph reducibility algorithm
强连通性算法	strong connectivity

十 五 画

撕裂法	diakoptics
de Bruijn序列	de Bruijn sequence
NP问题	NP problem
NP-完全	NP-complete
NP-难解	NP-hard